

Integration von Usability Engineering und Software Engineering:

Evaluation und Optimierung eines ganzheitlichen Modells
anhand von Konformitäts- und Rahmenanforderungen

MASTERTHESIS

ausgearbeitet von

Holger Fischer

zur Erlangung des akademischen Grades

MASTER OF SCIENCE (M.Sc.)

vorgelegt an der

FACHHOCHSCHULE KÖLN
CAMPUS GUMMERSBACH
FAKULTÄT FÜR INFORMATIK UND
INGENIEURWISSENSCHAFTEN

im Studiengang

MEDIENINFORMATIK

Erster Prüfer: Prof. Dr. Gerhard Pläßmann
Fachhochschule Köln

Zweiter Prüfer: Dr. Karsten Nebe
C-LAB, Universität Paderborn

Gummersbach, im Juli 2010

Adressen: Holger Fischer
Geringhausen 48
51588 Nümbrecht
studium@holgerfischer.info

Prof. Dr. Gerhard Plassmann
Fachhochschule Köln
Institut für Informatik
Steinmüllerallee 1
51643 Gummersbach
gerhard.plassmann@fh-koeln.de

Dr. Karsten Nebe
C-LAB, Universität Paderborn
Interactive Dialog Systems
Fürstenallee 11
33102 Paderborn
karsten.nebe@c-lab.de

Kurzfassung

Die Gebrauchstauglichkeit eines Produktes hat sich zu einem entscheidenden Qualitätskriterium in der Softwareentwicklung entwickelt. Neben den Nutzern erkennen auch immer mehr Organisationen den Mehrwert von gebrauchstauglichen Produkten, insbesondere die steigende Produktivität, die verbesserte Qualität und den damit verbundenen Profit.

Die Disziplin des Usability Engineering liefert Vorgehensmodelle, Methoden und Techniken zur Gestaltung gebrauchstauglicher Produkte. Durch eine ausgiebige Analyse des Nutzungskontextes und einer Spezifikation von Anforderungen auf Basis der Bedürfnisse der Benutzer, können Designalternativen gestaltet und unter Teilnahme zukünftiger Benutzer evaluiert werden.

Die Disziplin des Software Engineering beschäftigt sich mit Prinzipien, Methoden, Techniken und Werkzeugen zur Entwicklung, zur Qualitätssicherung, sowie zum Management großer Softwaresysteme.

In Praxis und Wissenschaft wird derzeit dem Wunsch nachgegangen, beide Disziplinen zu integrieren. Dabei werden die Aktivitäten des Usability Engineering meist dem bereits etablierten Software Engineering untergeordnet und häufig nur punktuell oder aber erst zu einem relativ späten Zeitpunkt in der Entwicklung durchgeführt. Der Einfluss auf die Gebrauchstauglichkeit des Produktes ist dadurch vergleichsweise gering.

Wie lassen sich also Aktivitäten beider Disziplinen so koordinieren, dass sie koexistent durchgeführt werden, Engpässe vermieden und doppelte Arbeit verhindert wird? Wie sollten Anknüpfungspunkte zwischen beiden Disziplinen aussehen?

Um diese Fragen zu beantworten, werden in dieser Arbeit Standards der beiden Disziplinen in Bezug auf Aktivitäten und Artefakte analysiert, welche in einem ganzheitlichen Modell zur Integration von Usability Engineering und Software Engineering abgebildet werden sollen. So sollen mögliche Anknüpfungspunkte zwischen Aktivitäten und Artefakten aufgezeigt werden.

Ziel ist die Evaluation dieses Modells anhand von Konformitäts- und Rahmenanforderungen, sowie durch eine Expertenbefragung.

Abstract

The usability of a product has become a significant quality criterion in the software engineering process. Not only the user, but also the organizations become aware of the benefit created by usable products, especially the rising productivity, the improved quality and therewith the higher profit.

The discipline of usability engineering provide process models, methods and techniques to design usable products. By analysing the context of use extensively and specifying requirements according to the user needs, design alternatives could be developed and evaluated with participation of future users.

The discipline of software engineering is engaged with principles, methods, techniques and tools for the development, quality assurance and management of huge software-systems.

Both practice and science deal with the integration of both disciplines. In the process the activities of usability engineering are subordinated to the well-established software engineering. Frequently the activities are selectively performed or they are carried out at a tardily point in time so that they have an minor impact to the usability of a product.

How can activities of both disciplines be coordinated in order to perform parallel, to avoid bottlenecks and to reduce tasks executed double time? How should points of contact between the disciplines look like?

In this thesis standards of both disciplines are analysed to answer these questions. Activities and artifacts are identified and mapped in a holistic model integrating usability engineering and software engineering. In this way points of contact between activities and artifacts should be identified.

The aim is the evaluation of this model in reference to conformity and frame requirements as well as by a questioning with experts.

Inhaltsverzeichnis

1. Einleitung	7
1.1. Motivation	7
1.2. Problemstellung	8
1.3. Aufgabenstellung	10
1.4. Zielsetzung	11
1.5. Struktur der Arbeit	11
2. Grundlagen	13
2.1. Disziplinen im Entwicklungsprozess	13
2.1.1. Usability Engineering	13
2.1.2. Software Engineering	21
2.1.3. Marketing und Marktforschung	26
2.2. Integration	27
2.2.1. Bedarf der Integration	28
2.2.2. Ebenen der Integration	29
2.2.3. Existierende Integrationsansätze	31
2.3. Standards	35
2.3.1. Nutzen von Standards	36
2.3.2. Standards im UE	37
2.3.3. Standards im SE	44
2.3.4. Folgerungen für diese Arbeit	47
2.4. Zusammenfassung	48
3. Integrationsmodell	49
3.1. Ausgangsbasis	49
3.1.1. Auswahl der Standards	50
3.1.2. Ermitteln der Aktivitäten und Artefakte	51
3.1.3. Digitalisierung der Ergebnisse	54
3.1.4. Intrinsische Validierung	62
3.2. Anpassungen	66
3.3. Validierung des Modells	69
3.3.1. Bedarf der Validierung	69
3.3.2. Konformitäts- und Rahmenanforderungen	69
3.3.3. Abgleich	70
3.4. Angepasstes Modell	71
3.5. Zusammenfassung	73
4. Expertenbefragung	76
4.1. Methodik	77

4.2. Durchführung	78
4.3. Resultate	83
4.4. Zusammenfassung	89
5. Zusammenfassung, kritische Würdigung und Ausblick	91
5.1. Zusammenfassung	91
5.2. Kritische Würdigung	93
5.3. Ausblick	96
Abbildungsverzeichnis	100
Tabellenverzeichnis	102
Literaturverzeichnis	103
ANHANG	112
A. Referenzen	113
B. Base Practices	131
C. Artefakt Lebenszyklus	143
D. Gemeinsamkeiten	146
E. Konformitäts- und Rahmenanforderungen	153
F. Expertenbefragung	163
G. Modelle	179
H. Glossar	201
Danksagung	213
Eidesstattliche Erklärung	214

1. Einleitung

Die vorliegende Arbeit beschäftigt sich mit dem Thema der Evaluation und Optimierung eines ganzheitlichen Modells zur Integration von Usability Engineering und Software Engineering.

In diesem Kapitel wird eine Motivation gegeben, warum dieses Thema betrachtet werden sollte. Im Anschluss wird die Problemstellung hergeleitet. Zudem wird die Aufgabenstellung beschrieben und die damit verbundene Zielsetzung dargelegt. Abschließend wird erläutert, wie diese Aufgaben und Ziele in der Struktur dieser Arbeit verankert wurden.

1.1. Motivation

Die Gebrauchstauglichkeit (englisch: *Usability*), als entscheidende Qualitätseigenschaft eines Produktes, gewinnt in der Entwicklung interaktiver Systeme zunehmend an Bedeutung. Der Begriff der Gebrauchstauglichkeit ist definiert als...

„...das Ausmaß, in dem ein Produkt durch bestimmte Benutzer in einem bestimmten Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und zufriedenstellend zu erreichen.“ (ISO 9241-11:1998)

Unter Effektivität wird dabei die Genauigkeit und Vollständigkeit verstanden, mit der ein Benutzer ein bestimmtes Ziel erreicht. Der im Verhältnis zur Genauigkeit und Vollständigkeit eingesetzte Aufwand wird durch die Effizienz ausgedrückt. Zufriedenstellung bezeichnet die Freiheit von Beeinträchtigungen beim Gebrauch eines Produktes, sowie die positive Einstellung gegenüber der Nutzung.

Gebrauchstaugliche Systeme sind in vielerlei Hinsicht erfolgreicher und stellen einen beachtlichen Gewinn sowohl für Benutzer, als auch für Arbeitgeber und Anbieter dar (vgl. ISO 9241-210:2010). Ausgehend von einer steigenden Produktivität der Benutzer werden betriebliche Abläufe der Organisationen effizienter. Trainings- und Supportkosten werden reduziert, da das Produkt einfacher zu verstehen und zu benutzen ist. Das Nutzungserlebnis (*UX*) wird verbessert und auch die Zugänglichkeit für Personen mit ausgeprägten Fähigkeiten wird erhöht (vgl. de Piotrowski & Tauber 2009).

Unannehmlichkeiten und Stress werden reduziert und damit das Gesundheits- und Sicherheitsrisiko, hinsichtlich der Benutzer, verringert. Auf Grund der genannten Faktoren entwickelt sich durch die Gebrauchstauglichkeit ein Wettbewerbsvorteil, der zum Beispiel auch das Markenimage der betreibenden beziehungsweise der herstellenden Organisation verbessern kann.

1.2. Problemstellung

Am Entwicklungsprozess gebrauchstauglicher Produkte sind zwei Disziplinen beteiligt beziehungsweise nehmen diese Einfluss auf den Prozess. Dies sind die Disziplinen des Usability Engineering (*UE*) und des Software Engineering (*SE*).

Usability Engineering stellt geeignete Vorgehensmodelle, Methoden und Techniken zur Verfügung, um ein möglichst hohes Maß an Gebrauchstauglichkeit in den entwickelten Systemen zu erlangen. Ausschlaggebend dafür ist eine detaillierte Analyse des Nutzungskontextes, sowie die Spezifikation von Anforderungen auf Basis der Bedürfnisse der Benutzer. So können verschiedene Designalternativen gestaltet und evaluiert werden. Zukünftige Benutzer des Systems sind dabei häufig im gesamten Prozess mit eingebunden. Dies hat vor allem Auswirkungen hinsichtlich einer Verbesserung der Interaktion zwischen Mensch und System, die sich in einer angemessenen Benutzerschnittstelle widerspiegelt.

Das Software Engineering liefert Prinzipien, Methoden, Techniken und Werkzeuge zur Entwicklung, zur Qualitätssicherung, sowie zum Management großer Softwaresysteme.

In größeren Unternehmungen existieren in der Regel etablierte Prozesse und Rollen für die Entwicklung von Systemen. Bereiche, Abteilungen, Aufgaben und Dokumente wurden definiert, sowie Rahmenbedingungen innerhalb von Projekten angelegt (vgl. Hennigs & Zimmermann 2009). Eine Herausforderung ist es, die Methoden, Aktivitäten und Artefakte des Usability Engineering in bestehende Strukturen von Organisationen einfließen zu lassen (vgl. Nebe 2009).

Aktivitäten des Usability Engineering werden dabei meist den bereits etablierten Prozessen des Software Engineering untergeordnet und häufig nur punktuell oder aber erst zu einem relativ späten Zeitpunkt in der Entwicklung durchgeführt. Der Einfluss auf die Gebrauchstauglichkeit eines Systems ist dadurch vergleichsweise gering. Um diesen zu verbessern, sollte das Usability Engineering bereits von Beginn des Projektes an tätig sein. Das Usability Engineering entwickelt besonders in den Phasen der Kontextanalyse und der Anforderungserhebung Modelle, wie zum Beispiel Benutzermodelle, Aufgabenmodelle, Kontextmodelle, die in der Phase der Evaluation essentiell zur Bewertung der Gebrauchstauglichkeit beziehungsweise zur Durchführung einiger Evaluationstechniken

sind.

Anhand der Vielzahl der in den letzten Jahren erschienenen Veröffentlichungen zum Thema der Integration, lässt sich erkennen, dass derzeit sowohl eine wachsende Akzeptanz des Aspektes der Gebrauchstauglichkeit, als auch die Notwendigkeit einer Integration des Usability Engineering mit dem Software Engineering besteht. Sowohl „[...] *in der Theorie als auch in der Praxis existieren Bestrebungen, entsprechende Lösungsansätze zu liefern.*“ (Nebe 2009)

Integration bedeutet dabei, die Interessen und Ziele beider Disziplinen gleichermaßen zu berücksichtigen und in einem großen Ganzen miteinander zu vereinen. Dabei sollten nach Nebe (2009) Integrationsaspekte auf drei Ebenen (Standards, Modelle, operativer Prozess) betrachtet werden. Existierende Ansätze spezifizieren zum Beispiel generische UE Aktivitäten (Ferre 2003), formulieren gemeinsame Artefakte (Seffah et al. 2005b) oder erweitern bestehende SE Vorgehensmodelle (Göransson et al. 2003).

Erwartet wird dabei der bereits im Abschnitt zur Motivation beschriebene Mehrwert: die Entwicklung und das Management von Softwaresystemen unter Berücksichtigung des durch das Usability Engineering hervorgebrachten Aspektes der Gebrauchstauglichkeit.

Des Weiteren würde durch eine Integration zugleich die Position des Usability Engineering in Organisationen gestärkt werden. In diesem Zusammenhang könnten gegebenenfalls viele Argumente bezüglich des Mehraufwandes und der Schwierigkeiten auf Grund heterogener Vorgehensmodelle wegfallen (vgl. Nebe 2009).

Einige Fragestellungen, die zur Motivation und Herausforderung am Thema beitragen und die es zu beantworten gilt, lauten wie folgt:

- Wie sollten konkrete Anknüpfungspunkte zwischen UE und SE aussehen?
- Wie lassen sich Rollen und Aktivitäten koordinieren, so dass Engpässe (englisch: *bottlenecks*) und doppelte Arbeit verhindert werden kann?
- Welche Überschneidungen von Rollen und Aufgaben zwischen UE Experten und anderen entstehen beziehungsweise existieren?
- Wie müssen geeignete Dokumentationen aussehen, um Anforderungen zu kommunizieren?

1.3. Aufgabenstellung

Bereits im Vorfeld dieser Arbeit wurden Phasen, Aktivitäten und Artefakte des Usability Engineering und Software Engineering identifiziert. Auf deren Basis sollte ein strukturelles Modell erstellt werden, welches die Aktivitäten und Artefakte beider Disziplinen nebeneinander abbildet. Ziel dabei war es mögliche Anknüpfungspunkte für die Integration aufzuzeigen und Aussagen über gemeinsame Aktivitäten und Artefakte zu treffen. Als Grundlage dafür wurden etablierte Standards verwendet, da diese einen anerkannten Konsens der Experten veranschaulichen. Zudem stellen Standards die oberste Integrationsebene dar, so dass noch keine Anpassung an einen spezifischen, organisatorischen Kontext beziehungsweise an die Komplexität einer Organisation stattgefunden hat (vergleiche dazu Kapitel 2.2.2). Das Modell existiert somit auf einer abstrakten Ebene und bleibt skalierbar. Für das Usability Engineering wurde daher die DIN EN ISO 13407 (1999) und für das Software Engineering die ISO/IEC 12207 (2002) betrachtet. Im Anschluss an die Erstellung wurde eine erste Beurteilung des Modells, in Bezug auf dessen Vollständigkeit, durchgeführt. Anhand von zwei Standards zur Bewertung von Prozessen sollte die intrinsische Validität des Modells nachgewiesen werden. Dazu wurde der Standard ISO/TS 18152 (2010) für das Usability Engineering und der Standard ISO/IEC 15504 (2006) für das Software Engineering verwendet. Beide Standards formulieren so genannte *Base Practices* und *Work Products*, die mit den Aktivitäten und Artefakten des Modells abgeglichen wurden. Des Weiteren wurde das Modell um Base Practices erweitert, die vorher nicht auf entsprechende Aktivitäten abgebildet werden konnten (vergleiche dazu Kapitel 3.1).

Auf Grund des mittlerweile neu veröffentlichten Standards ISO 9241-210 (2010), als Nachfolger der DIN EN ISO 13407, sieht die Aufgabenstellung nun vor, das existierende Modell dahingehend zu überarbeiten. So soll die Aktualität des Modells gewährleistet werden. Das heißt, dass Aktivitäten und Artefakte auf Basis des neuen Standards ermittelt werden, um die im Modell aus Perspektive des Usability Engineering bereits verankerten Elemente zu überarbeiten.

Des Weiteren findet ein Abgleich des Modells mit den Standards ISO/TS 18152 und ISO/IEC 15504 statt, um die intrinsische Validität des Modells erneut sicherzustellen. Das so aktualisierte Modell stellt dann eine geeignete Ausgangsbasis für die daran anschließende Evaluation des Modells dar.

Hauptaufgabe dieser Arbeit ist die extrinsische Evaluation des Modells, wodurch die Konstruktvalidität sichergestellt werden soll. Dazu wird das vorhandene Modell mit denen in der Dissertation von Nebe (2009) definierten und durch Experten validierten Konformitäts- und Rahmenanforderungen abgeglichen. Das bedeutet, dass das hier

entwickelte Modell gegebenenfalls um fehlende Aktivitäten ergänzt und optimiert wird. Ergebnis ist somit eine erweiterte Form des Modells.

Wichtig ist insgesamt der Bezug des Modells zur Praxis, um dessen Sinnhaftigkeit zu begründen. Daher soll das erweiterte Modell durch ein Interview mit Experten aus der Praxis evaluiert werden. Die Ergebnisse fließen entsprechend wieder in das Modell mit ein.

1.4. Zielsetzung

Ziel der Masterthesis soll sein, auf Abstraktionsebene von Standards eine detaillierte Basis in Form eines strukturellen Modells der Aktivitäten und Artefakte mit zusätzlichen semantischen Informationen zu schaffen. Dieses soll als weiterer Ansatzpunkt für eine Integration des Usability Engineering und Software Engineering dienen. Dabei sollen mögliche Anknüpfungspunkte und gemeinsame Aktivitäten und Artefakte der Disziplinen identifiziert werden.

Dies geschieht bewusst auf Ebene der Standards, damit das entstandene Modell möglichst skalierbar bleibt und je nach Projekt speziell auf dieses angewendet werden kann.

Des Weiteren soll die vorliegende Arbeit einen Beitrag leisten in Hinblick auf die Beantwortung der Frage: *„Wie können die Disziplinen des Usability Engineering und Software Engineering sinnvoll integriert werden, so dass die Interessen und Ziele beider Perspektiven gleichermaßen berücksichtigt werden?“*

1.5. Struktur der Arbeit

In **Kapitel 2** werden Grundlagen vermittelt, die von essentieller Bedeutung für das weitere Verständnis der Arbeit sind. Dazu wird zunächst das Usability Engineering und das Software Engineering betrachtet. Es wird beschrieben, welche Ziele diese Disziplinen verfolgen und welchen Mehrwert diese mit sich bringen. Daraufhin wird der Bedarf einer Integration verdeutlicht, indem die möglichen Ebenen einer Integration aufgezeigt und existierende Ansätze der Integration diskutiert werden. Da die im Rahmen eines Projektes vorausgegangene Entwicklung eines Integrationsmodells auf eine Integration auf Ebene der Standards abzielt, werden anerkannte Standards vorgestellt. Des Weiteren wird der allgemeine Nutzen von Standards begründet, sowie Folgerungen für diese Arbeit formuliert.

In **Kapitel 3** wird zunächst die Struktur und die Erkenntnisse des bereits im Vorfeld dieser Arbeit erstellten Modells erläutert, da es als Ausgangsbasis für diese Arbeit

dient. Da der UE-Standard DIN EN ISO 13407 im März 2010 vom Standard ISO 9241-210 abgelöst wurde, muss eine Anpassung des vorhandenen Modells stattfinden, deren Ergebnisse in diesem Kapitel zu lesen sind. Des Weiteren wird der Abgleich des Modells mit dem Standard ISO/TS 18152 dargestellt, da diese für die darauf folgende Evaluation notwendig ist. Dies stellt die Kernaktivität dieser Arbeit dar. Um die Konstruktvalidität des entwickelten Modells sicherzustellen, wird dieses mit Konformitäts- und Rahmenanforderungen an einen Softwareentwicklungsprozess abgeglichen. Diese Konformitäts- und Rahmenanforderungen beruhen auf der Dissertation von Nebe (2009), der diese erarbeitet, formuliert und durch Experten hat validieren lassen. Anhand der Resultate des Abgleiches entsteht ein angepasstes Modell, welches abschließend in diesem Kapitel betrachtet wird.

In **Kapitel 4** wird die durchgeführte Expertenbefragung vorgestellt. Diese soll weitere Erkenntnisse über die Validität des Modells aufzeigen. Dazu wird in diesem Kapitel die Methodik zur Formulierung der Fragen und die Durchführung beziehungsweise Auswertung der Befragung erläutert. Abschließend werden die ausgewerteten Antworten und daraus gewonnen Erkenntnisse interpretiert.

In **Kapitel 5** wird die Arbeit noch einmal zusammengefasst, bevor sie kritisch gewürdigt wird. Anschließend wird ein Ausblick auf anknüpfende Themen und offene Fragen gegeben und so die Arbeit abgeschlossen.

2. Grundlagen

In Anlehnung an die Problemstellung und Zielsetzung werden in diesem Kapitel Grundlagen erläutert, auf denen die Inhalte dieser Arbeit basieren. Zunächst werden daher die Ursprünge und Ziele des Usability Engineering und Software Engineering verdeutlicht, sowie der daraus entstehende Mehrwert betrachtet. Des Weiteren wird erörtert, warum das Interesse an einer Integration besteht und welche Aspekte es dabei zu betrachten gilt. So werden Ebenen aufgezeigt, die nach Nebe (2009) bei einer Integration berücksichtigt werden sollten. Zudem werden exemplarisch einige Integrationsansätze diskutiert.

Da das im Vorfeld dieser Arbeit erstellte Integrationsmodell (siehe Kapitel 3.1) auf eine Integration auf Ebene der Standards abzielt, werden anerkannte Standards beider Disziplinen vorgestellt, sowie der allgemeine Nutzen von Standards begründet. Welche Folgerungen sich daraus für diese Arbeit ergeben, ist abschließendes Thema dieses Kapitels.

2.1. Disziplinen im Entwicklungsprozess

Um die Gebrauchstauglichkeit von Produkten herzustellen, bedarf es entsprechender systematischer Vorgehensweisen im Entwicklungsprozess, die durch das Usability Engineering und Software Engineering bereitgestellt werden.

Im Folgenden werden daher die beiden Disziplinen näher erläutert. Dazu wird zunächst jeweils der historische Ursprung und die Entwicklung betrachtet und die daraus resultierenden Ziele, sowie deren Mehrwert verdeutlicht. Um die Ziele zu erreichen, werden von den Disziplinen unter anderem Vorgehensmodelle eingesetzt, von denen einige exemplarisch vorgestellt werden.

Des Weiteren wird die Disziplin „Marketing & Marktforschung“ ansatzweise betrachtet, auf die in Kapitel 3.1.3 referenziert wird.

2.1.1. Usability Engineering

Der Begriff des Usability Engineering wurde 1986 von der Digital Equipment Corporation (vgl. Good et al. 1986), in Anlehnung an den Arbeiten von Gilb, Bennett, Butler, Carroll und Rosson, geprägt. Diese verstanden darunter den Prozess der ökonomische

Erstellung eines Arbeitssystems, welches bestimmte Bedürfnisse erfüllt. Um dieses Ziel zu erreichen, hielten sie es für notwendig messbare Ziele der Gebrauchstauglichkeit zu spezifizieren, so dass entschieden werden konnte, ob ein System die Bedürfnisse erfüllt oder nicht.

Der Grundgedanke von Good et al. (1986) ist auch heute noch im Usability Engineering enthalten. So definiert Faulkner (2000) die Disziplin des Usability Engineering als ...

„[...] an approach to the development of software and systems which involves user participation from the outset and guarantees the efficacy of the product through the use of a usability specification and metrics.“

Neue Bedürfnisse ergaben sich vor allem anhand der Benutzer eines Systems. Auf Grund der Entwicklung von persönlichen Computern (*PC*) und der damit verbundenen Verbreitung von Hardware und Software, waren nicht mehr nur die Entwickler von Software selbst die Anwender, sondern der Computer wurde für jeden zugänglich. Im Zuge dieser Entwicklungen wurden somit die Erwartungen und Bedürfnisse der Benutzer in den Fokus gerückt und die Schnittstelle zwischen dem Menschen und dem System erhielt eine größere Bedeutung. (vgl. Rosson & Carroll 2002)

Geprägt wird das interdisziplinäre Usability Engineering durch verschiedene Basisdisziplinen, unter anderem der Kognitionspsychologie, der Experimentalpsychologie oder der Ethnographie.

Kognitionspsychologie ist eine Wissenschaft, die sich mit der menschlichen Wahrnehmung (visuell, auditiv, olfaktorisch, gustatorisch, taktil, kinästhetisch, vestibulär) und Kognition (z.B. das menschliche Gedächtnis, Lernen, Probleme lösen, Entscheidungen treffen, logisches Schlussfolgern) beschäftigt. So kann mit Hilfe der Kognitionspsychologie zum Beispiel Wissen darüber gewonnen werden, wie die menschliche Informationsverarbeitung abläuft und dem entsprechend die Interaktion über die Benutzerschnittstelle formuliert oder bewertet werden.

Die Experimentalpsychologie ist eine Wissenschaft, welche das menschliche Verhalten untersucht und so dem Usability Engineering empirische Methoden zur Verfügung stellt, um die Leistung der Benutzer (englisch: *user performance*) und deren Zufriedenstellung zu messen.

Soziale und kulturelle Aspekte können mittels der Ethnographie untersucht, analysiert und beschrieben werden. Im Usability Engineering können mit Hilfe dieser Methoden die Benutzer erforscht und festgelegt, sowie Anforderungen an die Gebrauchstauglichkeit definiert werden. (vgl. Mayhew 1999)

Ein wesentlicher Begriff im Zusammenhang mit dem Usability Engineering, welcher bereits im Namen selbst verwendet wird, ist der Begriff der „Gebrauchstauglichkeit“ (Definition in Kapitel 1.1, S. 7). Einhergehend mit der Definition des Begriffs anhand der ISO 9241-11 lassen sich die wesentlichen Ziele des Usability Engineering identifizieren. Bei der Herstellung eines Produktes, beispielsweise in Form einer Software, sollen die Benutzer und der Nutzungskontext in die Entwicklung einbezogen werden. Nach Vollendung des Produktes soll der Benutzer dann seine Ziele effektiv, effizient und zufriedenstellend erreichen.

Wesentliche Merkmale des Usability Engineering sind die frühe Analyse der Benutzer und des Nutzungskontextes, sowie die Spezifikation von Benutzeranforderungen. Diese sind ausschlaggebend für die Qualität des entwickelten Produktes und werden daher den technischen Anforderungen gleichgestellt (vgl. Nebe 2009). Nutzungskontext und Benutzeranforderungen werden in der Gestaltung der Mensch-System-Interaktion und der Benutzungsschnittstelle berücksichtigt. Zudem werden sie für die anschließende Bewertung des entstanden Systems als Kriterien zur Überprüfung verwendet.

Der sich daraus erhoffte Mehrwert wird von Mayhew (1999) und der ISO 9241-210 wie folgt formuliert:

- Steigende Produktivität
- Steigende operationale Effizienz
- Einfacher zu verstehen und zu benutzen
- Steigende Gebrauchstauglichkeit für Personen mit ausgeprägten Fähigkeiten
- Sinkende Kosten und Zeit für Benutzerschulungen
- Verbessertes Nutzungserlebnis
- Reduzierte Unannehmlichkeiten und Stress
- Verringerte Benutzerfehler
- Erhöhung der Genauigkeit von Dateneingaben und Dateninterpretationen
- Steigende Wettbewerbsfähigkeit
- Verbessertes Markenimage
- Verringerte Entwicklungs- und Wartungskosten

- Reduzierte Kosten für Kundenunterstützung
- Förderung der Nachhaltigkeit

Um die Ziele des Usability Engineering und den damit verbundenen Mehrwert zu erreichen, bedarf es systematischer Vorgehensweisen. Daher ergibt sich auch der Begriff des „Engineering“ im Namen (vgl. Karat & Karat 2003). Im Laufe der letzten zwanzig Jahre wurden hierzu verschiedene Vorgehensmodelle entwickelt.

Ein Vorgehensmodell beschreibt einen Rahmen, in dem bestimmte Entwicklungsprozesse organisiert werden. Nach Zuser et al. (2004) ist ein Vorgehensmodell gleichzusetzen mit einer Projektdurchführungsstrategie. In dieser wird die Art und Weise festgelegt, wie Arbeitsschritte angeordnet sind und durchlaufen werden.

So kann ein Entwicklungsprozess in aufeinander folgende Phasen zerlegt werden. Eine Phase bezeichnet einen zeitlichen Abschnitt innerhalb eines Projektes. Dieser ist sachlich gegenüber den anderen Abschnitten getrennt, setzt sich aus darin stattfindenden Aktivitäten zusammen und schließt mit einem Meilenstein ab, welcher an einen Termin und ein bestimmtes Ergebnis geknüpft ist.

Die innerhalb einer Phase stattfindenden Aktivitäten sind in sich geschlossene Abfolgen von (Teil-)Tätigkeiten, die von wenigen Personen in einer gewissen Zeit durchführbar sind und ein festgelegtes Ergebnis liefern. Häufig bedarf es zur Durchführung einer Aktivität eines bestimmten Artefaktes als Ausgangsbasis. Erzielte Ergebnisse werden daraufhin entweder in einem neuen Artefakt festgehalten oder in das Artefakt der Ausgangsbasis eingearbeitet.

Bei einem Artefakt handelt es sich, wie bereits angedeutet, um ein Arbeitsergebnis, welches durch Ausübung einer Aktivität entsteht. Dieses besteht zum Beispiel aus einem Dokument, aus einem Quellcode oder aus einer Gestaltungslösung. Artefakte sind dynamisch und können sich im Laufe eines Projektes verändert, wachsen oder phasenübergreifend weiter entwickelt werden.

Des Weiteren existieren Rollen, um Verantwortlichkeiten und Zuständigkeiten in Bezug auf Aktivitäten und Artefakte in einem Projektteam zu organisieren. (vgl. Zuser et al. 2004; Winter 2005)

Im Folgenden werden exemplarisch drei Vorgehensmodelle vorgestellt.

Usability Engineering Lifecycle In ihrem Buch „The Usability Engineering Lifecycle“ beschreibt Mayhew (1999) ein Vorgehensmodell, welches einen kompletten Prozess, aus-

gehend von einem bereits definierten, geplanten und abgegrenzten Entwicklungsprojekt bis hin zur Installation, darstellt (siehe Abbildung 2.1).

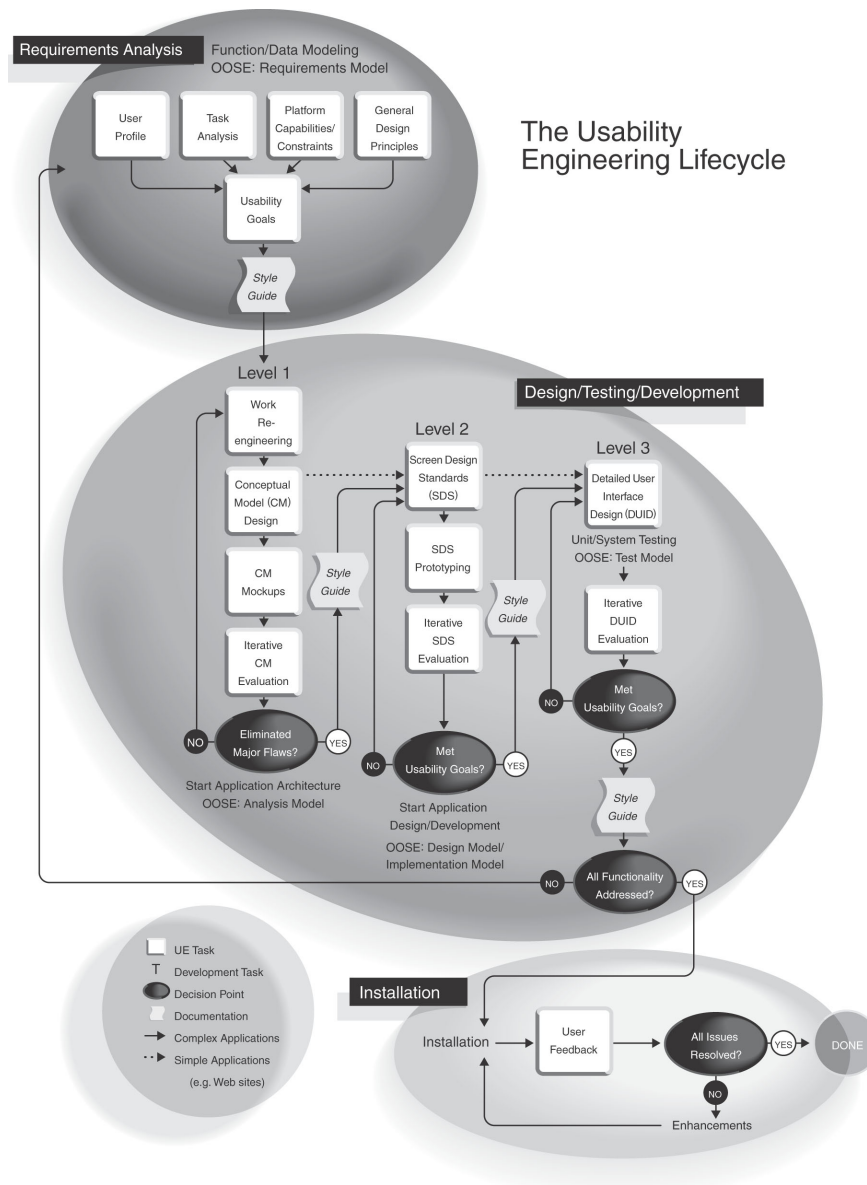


Abbildung 2.1.: Usability Engineering Lifecycle nach Mayhew (1999)

Das Vorgehensmodell ist unterteilt in drei Phasen und formuliert Aufgaben an den Usability Engineer. In der Phase der Anforderungsanalyse (*Requirements Analysis*) werden die Benutzer und deren Aufgaben, sowie technische Fähigkeiten und Einschränkungen durch das System analysiert. Zusammen mit generellen Designprinzipien werden die gewonnenen Erkenntnisse als Ziele der Gebrauchstauglichkeit spezifiziert und in einer Produktgestaltungsrichtlinie (*Product Style Guide*) festgehalten. Die zweite Phase der

Gestaltung, des Testens und der Entwicklung (*Design/Testing/Development*) ist gegliedert in drei Ebenen, die jeweils iterativ gestaltet sind. Auf der ersten Ebene werden die Aufgaben und der Arbeitsablauf der Benutzer auf Basis der Ergebnisse der Anforderungsanalyse neu gestaltet und in einem konzeptuellen Modell (*Conceptual Model*) spezifiziert, evaluiert und in einer Gestaltungsrichtlinie festgehalten. Diese ist Ausgangsbasis der zweiten Ebene, auf welcher Standards des *Screendesigns* betrachtet und Prototypen erstellt werden. Wie in jeder Ebene werden auch hier die Ergebnisse in einer Gestaltungsrichtlinie formuliert. Auf der dritten Ebene wird anhand der bisherigen Erkenntnisse die Benutzerschnittstelle gestaltet und evaluiert, sowie der Gestaltungsrichtlinie hinzugefügt. Sofern alle zuvor beschriebenen Funktionalitäten erfüllt werden, kann mit der Installation des Systems begonnen werden.

Das Modell von Mayhew (1999) ist skalierbar und kann unabhängig von der Zeit, dem Budget oder der Komplexität an das jeweilige Projekt angepasst werden. So unterstützt es Iterationen während der Gestaltungsphase, die jedoch übersprungen werden können. Nachteile im Modell bestehen darin, dass die Gestaltungsentwürfe nicht von technischen Gegebenheiten dekorreliert sind. Des Weiteren wird der Nutzungskontext nur ansatzweise in Form der Benutzerprofile (*User Profiles*) und der Aufgabenanalyse (*Task Analysis*) berücksichtigt. Die geplanten Evaluationen werden lediglich in Form eines *Usability Testing* auf Grundlage der Gestaltungsrichtlinien durchgeführt und schließen so zunächst eine Beteiligung der Benutzer aus. Rückmeldungen der Benutzer werden erst nach der Installation des fertigen Systems eingeholt, um dieses später verbessern zu können, wodurch das zuvor entwickelte System nicht unbedingt den Bedürfnissen der Benutzer entsprechen muss.

Eine Integration mit dem Software Engineering findet in der Hinsicht statt, dass gewonnenen Ergebnisse in Form der Gestaltungsrichtlinien dem Software Engineering für die Entwicklung zur Verfügung gestellt werden.

Scenario-based Usability Engineering Rosson & Carroll (2002) beschreiben in ihrem Buch „Usability Engineering“ einen auf Szenarien basierenden Ansatz, wie in Abbildung 2.2 dargestellt.

Dieser setzt sich aus drei jeweils iterativen Phasen, der Analyse, dem Design und dem Erstellen von Prototypen und der Evaluation zusammen.

In der Phase der Analyse wird zunächst ein Konzept erstellt, welches die Vision, eventuelle Annahmen und die Stakeholder beinhaltet. Mittels einer Feldforschung (*Field Studies*) werden Beobachtungen, Aufzeichnungen und Interviews durchgeführt, sowie bestehende Artefakte analysiert. Gewonnene Erkenntnisse über Stakeholder, Aufgaben, Artefakte und allgemeine Themen werden dokumentiert. Durch Problemszenarien wird

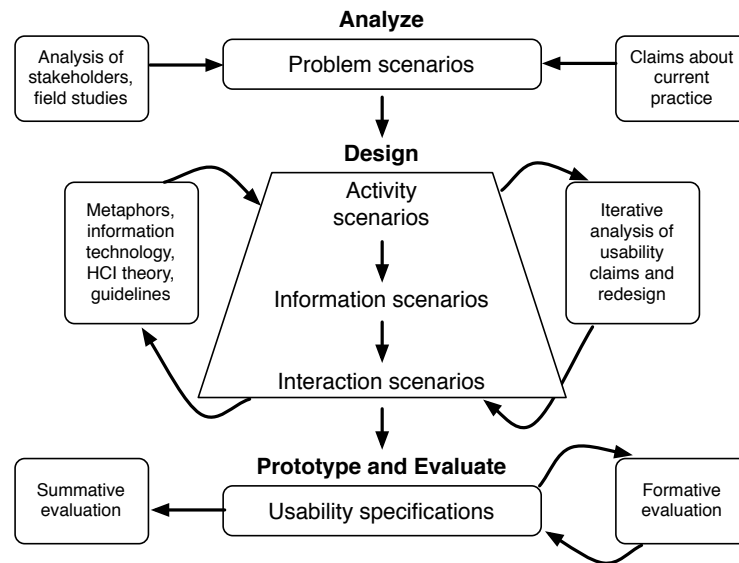


Abbildung 2.2.: Scenario-based Framework nach Rosson & Carroll (2002)

die aktuelle Situation erfasst und zuvor dokumentierte Ergebnisse in einen Kontext gesetzt. Die in den Problemszenarien formulierten Behauptungen müssen dann durch eine Analyse (*Claim Analysis*) validiert werden, um Folgerungen für den Gebrauch abzuleiten.

Nachdem die aktuelle Situation erfasst wurde, können in der zweiten Phase Lösungen entwickelt werden. Dies geschieht durch die Erstellung von Szenarien drei unterschiedlicher Typen. Im ersten Schritt werden Aktivitätsszenarien erstellt, welche die Erwartungen der Benutzer an das System beschreiben. Sie repräsentieren einen Blick auf das zukünftige System, welches realisiert werden soll. Aktivitätsszenarien werden weiterentwickelt zu Informationsszenarien, welche einen Überblick über die vom System dem Benutzer angebotenen Informationen ermöglichen sollen. Im dritten Schritt werden Interaktionsszenarien erstellt, welche detailliert die Aktionen der Benutzer, deren unterstützte Aufgaben, die benötigten Informationen und die Rückmeldungen vom System beschreiben.

Die in der Designphase erstellten Szenarien werden iterativ analysiert, Annahmen für die Gebrauchstauglichkeit getroffen und gegebenenfalls neu gestaltet. Dabei werden Metaphern, Möglichkeiten der Informationstechnologie, Theorien der Mensch-Computer-Interaktion und Gestaltungsrichtlinien berücksichtigt.

Annahmen aus der Designphase werden letztendlich in der dritten Phase mit Hilfe von Prototypen abgebildet und können so überprüft werden. Dabei werden Gestaltungslösungen kontinuierlich durch eine formative Evaluation bewertet und verbessert, um

abschließend mittels einer summativen Evaluation das fertige System zu verifizieren.

Ebenso wie das Modell von Mayhew (1999), handelt es sich auch bei dem Modell von Rosson & Carroll (2002) um ein iteratives Vorgehensmodell. Da die Erstellung der Szenarien und deren Analyse durch Mitglieder des Gestaltungsteams durchgeführt werden, werden auch hier die realen Benutzer nicht unbedingt mit in den Entwicklungsprozess einbezogen. Durch die frühe Berücksichtigung technischer Möglichkeiten, kann auch in diesem Modell nicht zwischen Mensch und System dekorreliert werden, so dass die gestalterische Sicht durch technische Bedingungen eingeschränkt wird.

Eine Integration mit dem Software Engineering wird durch die Bereitstellung von *Usability Specifications* ermöglicht.

Goal-Directed Design In ihrem Buch „About Face 3“ beschreiben Cooper et al. (2007) einen zielorientierten Gestaltungsprozess, den sie in sechs Phasen unterteilen (siehe Abbildung 2.3).

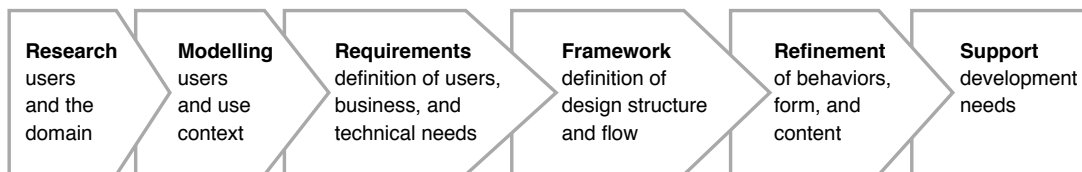


Abbildung 2.3.: Goal-Directed Design Process nach Cooper et al. (2007)

Im Modell werden diverse Techniken miteinander kombiniert. So finden Techniken der Ethnographie, Interviews mit Stakeholdern, Marktanalysen, Benutzermodelle, szenario-basiertes Design, sowie Prinzipien und Muster der Interaktionsgestaltung Verwendung. Erstellte Lösungen erfüllen so neben organisatorischen und technischen Imperativen, vor allem auch die Bedürfnisse und Ziele der Benutzer. Cooper et al. (2007) verzichten in ihrem Modell jedoch auf eine Iteration und evaluieren ihre Ergebnisse lediglich in der Phase der Verfeinerung (*Refinement*). Auch die Benutzer nehmen nur zu Beginn, in Form von Interviews, am Entwicklungsprozess teil.

Eine Integration mit dem Software Engineering findet höchstens durch eine Bereitstellung der Ergebnisse und durch eine Unterstützung des Entwicklungsteams statt.

Zusammenfassend liegt der Fokus des Usability Engineering auf der systematischen Bereitstellung von Vorgehensmodellen und Methoden zur Gewährleistung der Gebrauchstauglichkeit bei der Entwicklung interaktiver Systeme. Ausschlaggebend dabei ist die frühe Einbeziehung der Benutzer in den Entwicklungsprozess, um den Kontext, die Bedürfnisse und die daraus resultierenden Anforderungen zu erfassen. Diese werden

als Grundlage für eine kontinuierliche Bewertung der Lösungen verwendet. Neben dem Usability Engineering existiert das schon länger bestehende Software Engineering, auf das im folgenden Abschnitt eingegangen wird.

2.1.2. Software Engineering

Im Verlauf der Geschichte der Softwareentwicklung fand eine Software ihre Anwendung zunächst in den Naturwissenschaften. Zu diesem Zeitpunkt waren die Anwender einer Software hauptsächlich Wissenschaftler, die im Rahmen ihrer Forschungen komplexe Berechnungen mit Hilfe des Computers durchführen wollten und daher ihre Software selbst entwickelten. (vgl. Balzert 2001)

Mit der Entwicklung des persönlichen Computers veränderte sich der Kreis der Personen die an der Entwicklung und Benutzung einer Software beteiligt waren. Der Bedarf an Software, sowie deren Komplexität stieg. Damit einhergehend veränderten sich auch die Anforderungen an eine Software. Diese beinhalteten Zuverlässigkeit, Änderbarkeit, Übertragbarkeit, Effizienz, etc.. In Folge der veränderten Anforderungen und der steigenden Komplexität kam es in den späten 1960er Jahren zu der so genannten „Softwarekrise“. Dabei überstiegen die Kosten für Software die Kosten für Hardware und es kam zu den ersten großen gescheiterten Projekten, auf Grund von Zeit- und Kostenüberschreitungen. (vgl. Winter 2005)

Auf einer NATO-Konferenz im Jahre 1969 wurde daraufhin der Begriff des „Software Engineering“ geprägt (vgl. Buxton & Randell 1969). In Anlehnung an die Ingenieurwissenschaften soll das Software Engineering Prinzipien, Methoden, Techniken und Werkzeuge zur Entwicklung großer Softwaresysteme zur Verfügung stellen. Ziel ist die kostengünstige Produktion von Software mit einem festgelegten Funktionsumfang, sowie einer hohen Qualität und einem festen Budget- und Zeitrahmen. Neben der eigentlichen Softwareentwicklung beinhaltet das Software Engineering somit auch die Teildisziplinen der Qualitätssicherung und des Managements. Auf Grund der an der Softwareentwicklung beteiligten meist heterogenen Entwicklerteams müssen auch Themen, wie „interdisziplinäres Wissen“ und „soziale Kompetenz“, berücksichtigt werden. (vgl. Winter 2005)

In offiziellen Dokumenten, wie dem „Guide to the Software Engineering Body of Knowledge“ des IEEE Computer Society Professional Practices Committee (SWEBOK 2004), konnte mittlerweile das Software Engineering konsolidiert und in folgende Wissensgebiete gegliedert werden (vgl. Winter 2005):

- Anforderungsermittlung (*Software Requirements*)

- Entwurf (*Software Design*)
- Realisierung (*Software Construction*)
- Testen (*Software Testing*)
- Wartung (*Software Maintenance*)
- Konfigurationsmanagement (*Software Configuration Management*)
- Projektmanagement und Metriken (*Software Engineering Mangement*)
- Vorgehensmodelle (*Software Engineering Process*)
- Werkzeuge und Methoden (*Software Engineering Tools and Methods*)
- Softwarequalität (*Software Quality*)

Im Software Engineering bilden Vorgehensmodelle die Basis für einen systematischen und kontrollierten Ablauf der Entwicklung von Software. Exemplarisch werden im Folgenden drei Vorgehensmodelle vorgestellt.

V-Modell Das V-Modell wird bildhaft als „V“ dargestellt, wie in Abbildung 2.4 ersichtlich, und basiert auf den Entwicklungsstufen des Wasserfallmodells (linke Seite). Dieses wurde um die Softwarespezifizierung und um wichtige Schritte der Qualitätssicherung (rechte Seite) erweitert, wobei die Qualität im Wasserfallmodell erst zum Ende des Prozesses überprüft wird. (vgl. Winter 2005)

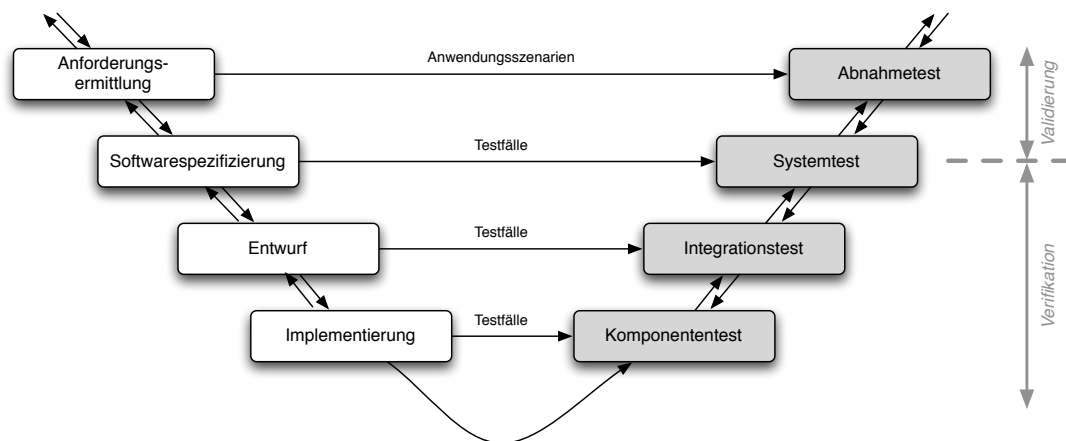


Abbildung 2.4.: V-Modell nach Balzert (2008)

Während auf der linken Seite des Modells vorwiegend analytische beziehungsweise konkretisierende Aktivitäten durchgeführt werden, beschreibt die rechte Seite des Modells

verifizierende beziehungsweise validierende Aktivitäten. Auf Basis der in den Aktivitäten der linken Seite erstellten Artefakte werden Testfälle definiert, welche für die Aktivitäten der rechten Seite verwendet werden (siehe die horizontalen Verbindungen im Modell).

Beim V-Modell handelt es sich um ein iteratives Vorgehensmodell. Iterationen wurden im ursprünglichen Wasserfallmodell nicht berücksichtigt. Im Laufe der letzten Jahre wurde das hier abgebildete V-Modell um die Versionen des V-Modell '97 und V-Modell XT erweitert. So wurden mittlerweile auch Entwicklungsansätze, wie zum Beispiel die Objektorientierung, betrachtet und das Modell skalierbarer gestaltet. (vgl. Nebe 2009) Des Weiteren konnte durch das Modell die Wichtigkeit aufgezeigt werden in einem frühen Stadium mit dem Überprüfen der Ergebnisse anzufangen. So können Fehler bereits frühzeitig, zum Beispiel in der Spezifikation, verhindert werden, wodurch der anschließende Aufwand der Pflege und Wartung des Systems verringert wird.

Rational Unified Process Bei dem Rational Unified Process (*RUP*) handelt es sich um ein iteratives Vorgehensmodell (siehe Abbildung 2.5), welches in Bezug auf objektorientierte Entwicklungsansätze gestaltet wurde. Im Modell werden neun so genannte Disziplinen betrachtet, welche eine Ansammlung von thematisch strukturierten Tätigkeiten darstellen. (vgl. Kruchten 2003)

Auf der zeitlichen Achse befinden sich vier aufeinander folgende Phasen. In der Konzeption (*Inception*) werden im Wesentlichen die grundlegenden Anforderungen an das System und der Umfang des Projektes ermittelt. In der darauf folgenden Phase der Ausarbeitung (*Elaboration*) wird die Anforderungsermittlung fortgeführt, so dass nach Abschluss dieser Phase eine möglichst stabile Spezifikation vorliegt. Des Weiteren wird der Systementwurf weitestgehend spezifiziert und in einem ersten Prototyp bezüglich der technischen Risiken des Projektes evaluiert. Eine Implementierung der Funktionalitäten, strukturiert in parallel entwickelbaren Modulen, und umfangreiche Tests sind Inhalt der Konstruktionsphase (*Construction*). In der abschließenden Phase des Übergangs in den Betrieb (*Transition*) erfolgt die Installation des Systems beim Anwender, sowie der zuvor durchgeführte „Beta-Test“ mit einer ausgewählten Benutzergruppe. (vgl. Winter 2005)

In diesem Modell werden besonders die parallel durchzuführenden Disziplinen betont. So sind zwar in jeder Phase gewisse Meilensteine, in Form von zum Beispiel Artefakten, definiert, die aber im gesamten Prozess geändert und verbessert werden können. Kontinuierlich geplante Tests und Iterationen sollen genau wie im V-Modell die Qualität des Systems sicherstellen.

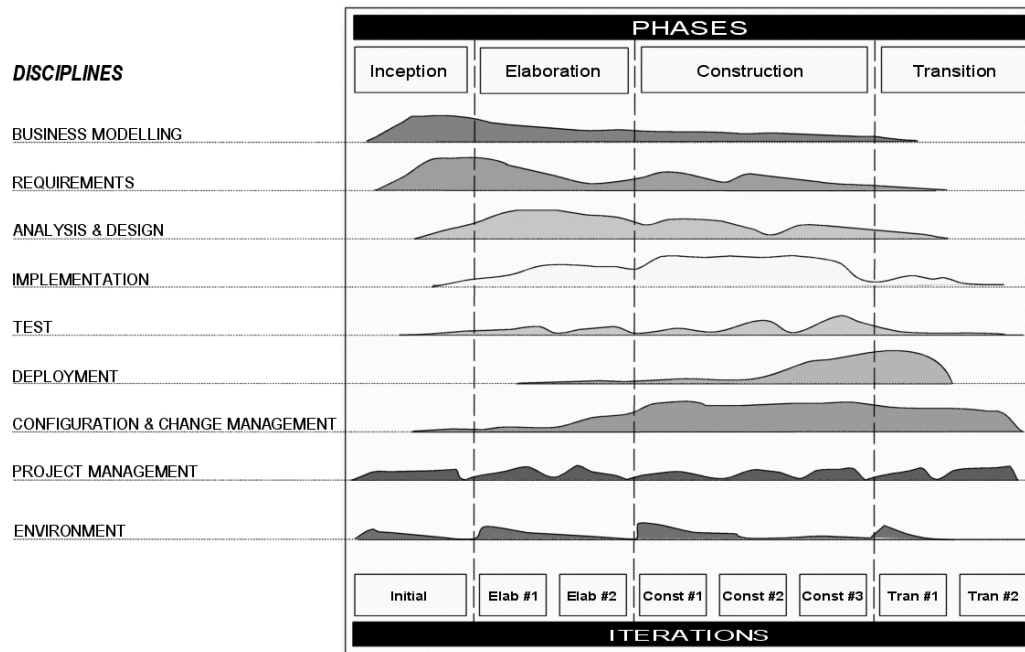


Abbildung 2.5.: Rational Unified Process nach Kruchten (2003)

Scrum Das Scrum-Modell (Abbildung 2.6) repräsentiert ein iteratives, agiles Vorgehensmodell, in welchem der Fokus, ähnlich wie beim V-Modell, eher auf dem Management als auf der Entwicklung eines Systems liegt.

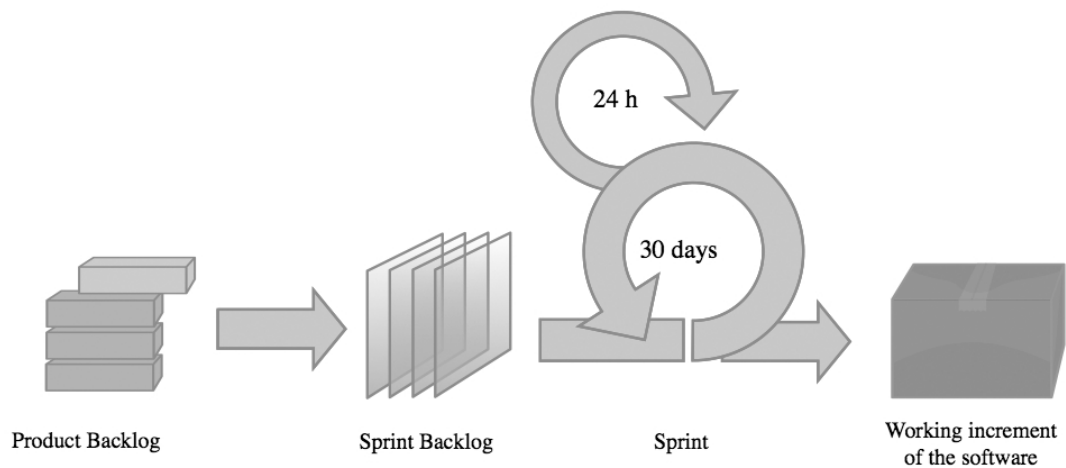


Abbildung 2.6.: Scrum nach Staudacher & Langenbacher (2008)

Der Auftragsbestand des Produktes (*Product Backlog*) umfasst die priorisierte Gesamtheit aller Anforderungen, Funktionen, Optimierungsvorschläge und Fehler. Ausgehend davon werden in einem so genannten *Sprint Planning Meeting* die einzelnen *Sprints* des Projektes geplant. Dazu wird ein Teil des Product Backlogs in einen *Sprint Backlog*, den Auftragsbestand für einen Sprint, übertragen. Unter dem Begriff *Sprints* werden Iterationen verstanden, in denen die Tätigkeiten der Analyse, des Designs, der Implementierung und der Dokumentation stattfinden. Ein Sprint umfasst in etwa 30 Tage und hat die Abarbeitung des Sprint Backlogs zum Ziel. Am Anfang eines jeden Tages entscheidet das *Sprint Team* über das jeweilige Tagesziel und diskutiert die Ergebnisse des Tages zuvor. Am Ende eines Sprints werden die Ergebnisse dann dem Kunden (*Product Owner*) im so genannten *Sprint Review* präsentiert und zur Diskussion gestellt, bevor in einem anschließenden Sprint Planning Meeting der nächste Sprint geplant wird. In einem 30-tägigen Sprint arbeitet das Team somit ungestört auf einer definierten, sich nicht ändernden Anforderungsmenge, während sich die Anforderungen im Product Backlog mitunter täglich ändern können, aber jeweils erst im nächsten Sprint berücksichtigt werden. Das eigentliche Produkt wächst dabei von Sprint zu Sprint inkrementell. (vgl. Staudacher & Langenbacher 2008)

Das agile Scrum-Modell bietet eine Alternative zu klassischen Vorgehensmodellen, wie etwa dem V-Modell oder dem RUP. Dem Entwicklerteam kommt durch dieses Modell eine besondere Verantwortung zu, da sie sich in einem Sprint selbst organisieren müssen und ihre Arbeit selbst definieren. Da das finale Produkt inkrementell wächst und nach jedem Sprint dem Kunden im Review präsentiert wird, kann das Risiko einer Entwicklung, die nicht die Bedürfnisse und Ziele der Nutzer erfüllt, verringert werden (vgl. Staudacher & Langenbacher 2008).

Da der Fokus im Modell auf dem Management eines Projektes liegt, besitzt das Modell Möglichkeiten der Integration von anderen Methoden, zum Beispiel aus dem Bereich des Usability Engineering (vgl. Nebe et al. 2007).

Zusammenfassend lässt sich festhalten, dass der Fokus des Software Engineering auf der systematischen Bereitstellung von Prinzipien, Methoden, Techniken und Werkzeugen zur Entwicklung und zum Management großer Softwaresysteme liegt, unter Berücksichtigung der Faktoren Zeit und Budget. Besonders der Aspekt der kontinuierlichen Qualitätssicherung spielt dabei eine zentrale Rolle in aktuellen Vorgehensmodellen der Disziplin.

2.1.3. Marketing und Marktforschung

Kotler et al. (2006, S. 30) definiert den Begriff des Marketing wie folgt:

„Marketing ist ein Prozess im Wirtschafts- und Sozialgefüge, durch den Einzelpersonen und Gruppen ihre Bedürfnisse und Wünsche befriedigen, indem sie Produkte und andere Dinge von Wert erzeugen, anbieten und miteinander austauschen.“

Der Definition entsprechend ergeben sich fünf Bausteine des Marketing-Konzeptes, die aufeinander aufbauen (siehe Abbildung 2.7): *Bedürfnisse, Wünsche und Nachfrage*; *Produkte, Dienstleistungen und Erlebnisse*; *Kundennutzen und Zufriedenheit*; *Austausch, Transaktion und Handelsbeziehungen*; *Märkte*.

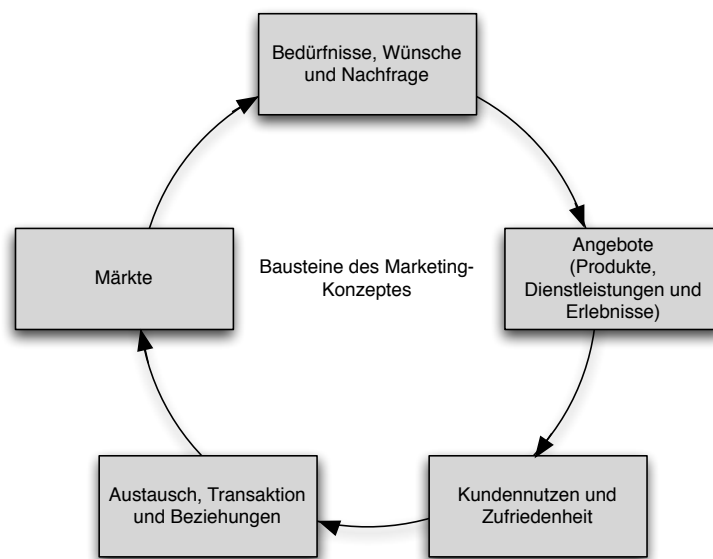


Abbildung 2.7.: Bausteine des Marketing-Konzeptes nach Kotler et al. (2006)

Betrachtet man nun Ziele des Marketing, so lässt sich feststellen, dass sich einige dieser Ziele mit denen einer gebrauchstauglichen Gestaltung von Produkten decken (vgl. Kotler et al. 2006, S. 51ff, 71ff, 537ff, 553ff):

- Maximale Lebensqualität
- Maximale Verbraucherzufriedenheit
- Kundenzufriedenheit
- Wertentstehung

- Kundennutzen
- Kundenbindung

Eine weitere wichtige Aktivität im Marketing-Informationssystem (Kotler et al. 2006, S. 404ff) zur Erreichung der zuvor genannten Ziele und Bausteine, ist die *Informationsbeschaffung und -aufarbeitung*. Da die Verantwortlichen im Marketing weder die Zeit noch die Mittel haben, um sämtliche nötigen Informationen selbst zu beschaffen, bedienen sie sich der Methoden der Marktforschung. Laut Kotler et al. (2006) stellt die Marktforschung die Verbindung von der Marketingabteilung zu den Verbrauchern, den Kunden und zur Öffentlichkeit her.

Zusammenfassend lässt sich festhalten, dass im Marketing und in der Marktforschung einige Aktivitäten existieren, welche die Analyse des Nutzungskontextes im Usability Engineering um weitere Informationen erweitern könnten.

2.2. Integration

Im vorherigen Abschnitt wurden das Usability Engineering und das Software Engineering mit ihren Zielen und Vorgehensmodellen, das Mittel zur Zielerreichung, vorgestellt. Anhand des Themas ersichtlich, behandelt diese Arbeit eine Möglichkeit zur Integration dieser beiden Disziplinen. Daher wird in diesem Abschnitt der Fragen nachgegangen, was Integration überhaupt bedeutet, warum Integration als geeignetes Paradigma gilt, warum der Bedarf einer Integration immer größer wird und wie die Integration der Disziplinen gestaltet werden kann.

Der Begriff der Integration lässt sich aus dem lateinischen Verb *integrare* („zurechtbringen, einrenken, von neuem Beginnen“) und vom lateinischen Attribut *integre* („unversehrt, vorurteilsfrei, unbefangen, unparteiisch“) ableiten. Des Weiteren findet sich im Lateinischen und Griechischen die Übersetzung „Herstellung eines Ganzen“ und „alle Aspekte der Ganzheitsbildung“. Im Fremdwörterbuch des deutschen Duden Verlags lautet die Definition „Wiederherstellen einer Einheit aus Differenziertem“ und „Eingliederung in ein großes Ganzes“.

Übertragen auf die Integration der Disziplinen des Usability Engineering und Software Engineering bedeutet dies, dass die Interessen und Ziele beider Perspektiven gleichermaßen und objektiv, also frei von Vorurteilen, in einem großen Ganzen eine Einheit bilden sollten. Soweit dies unter dem Begriff der „Integration“ verstanden wird, kann dieser als geeignetes Paradigma für die Zusammenführung der Disziplinen betrachtet werden.

2.2.1. Bedarf der Integration

Die Gebrauchstauglichkeit wird mittlerweile als Qualitätseigenschaft und als strategisches Mittel zur Abgrenzung gegenüber Konkurrenten von den Organisationen wahrgenommen (vgl. Ferre 2003). Dies resultiert vor allem an denen in Kapitel 2.1.1 (Seite 15) genannten Vorteilen, wie zum Beispiel den verringerten Entwicklungs- und Wartungskosten. Neben den zahlreichen Vorteilen für die Organisationen, sprechen auch viele Faktoren für die Gebrauchstauglichkeit auf Seiten der Benutzer (Beispiele siehe Kapitel 2.1.1).

Trotz des aktuellen Bewusstseins für die Gebrauchstauglichkeit bei Softwareentwicklungsteams, verbleibt diese jedoch im Tätigkeitsgebiet von Visionären, vereinzelt Abteilungen, aufgeklärten Softwareentwicklern und großen Organisationen, anstatt in der alltäglichen Praxis von Softwareentwicklern aufgenommen zu werden (vgl. Seffah et al. 2005b).

Die Eigenschaft der verbesserten Qualität bezieht sich dabei nicht nur auf das entwickelte Produkt, sondern gilt gleichermaßen für den Prozess der Entwicklung. So verfolgt das Usability Engineering das Ziel, dass *„[...] durch den Prozess der Erstellung sichergestellt werden [kann], dass die Lösung eine angemessene Qualität aufweist“* (Nebe 2009).

Daraus resultiert, dass nicht nur einzelne Methoden und Techniken des Usability Engineering in das Software Engineering eingegliedert werden sollten, da das Usability Engineering so nur Einfluss auf Teilprozesse, aber nicht auf den gesamten Prozess, ausüben kann.

Seffah et al. (2005b) argumentieren damit, dass ein integriertes Rahmenwerk, welches die Prinzipien von Analyse & Design, Entwicklung und Evaluation von beiden Perspektiven miteinander vereint, zu einer gesteigerten Effektivität von Usability Engineering in der Softwareentwicklung führt. Daher untersuchen aktuell mehrere Forschungsarbeiten, in wie weit Usability Engineering und Software Engineering miteinander integriert werden können (siehe dazu Kapitel 2.2.3).

Eine Integration der beiden Disziplinen scheint jedoch schwieriger als gedacht (vgl. Ferre 2003). Dies ist besonders auf einige Missverständnisse und Hindernisse zurückzuführen, die von Seffah & Metzker (2004) identifiziert wurden und von denen hier exemplarisch drei gelistet werden.

Die Bedeutung von Usability: Aus den Anfängen der Mensch-Computer-Interaktion haben sich einfache Bedeutungen, wie „einfach zu benutzen“ oder „benutzerfreundlich“, gefestigt, so dass Usability Engineering oft als „Oberflächenmaniküre“ der Benutzerschnittstelle gesehen wird. Dass noch andere Faktoren, wie zum

Beispiel der Nutzungskontext (ISO 9241-11), ausschlaggebend sind und Usability Engineering mehrere Methoden und Techniken anbietet, ist den Entwicklern häufig nicht bewusst.

„The People Gap“: Den Erfahrungen von Seffah & Metzker (2004) nach, sind viele UE Experten Psychologen, die aus Sicht der Softwareentwickler eher „im Weg stehen“. Dies liegt vor allem daran, dass beide Personengruppen nicht die selbe Kultur teilen und nicht über die Rahmenbedingungen der jeweils anderen Gruppe aufgeklärt sind. Während das Software Engineering eine allgemeine, systemgetriebene Philosophie fokussiert, werden im Usability Engineering eher soziale, kognitive und interaktionsbezogene Phänomene betrachtet (vgl. Sutcliffe 2005). So hat sich gezeigt, dass UE Methoden von UE Experten, die gleichzeitig Programmierer und Analysten sind, wesentlich besser von SE Experten akzeptiert werden und Einfluss im Entwicklungsprozess finden (vgl. Seffah & Metzker 2004).

Entbehrlichkeit von UE: Projektmanager sehen häufig nur die zeitlichen Faktoren, wenn UE Experten mehrere Iterationen durchführen um alles „perfekt“ zu gestalten. Dies liegt hauptsächlich am Fehlen deutlicher Beziehungen und Zusammenhänge zwischen Usability Engineering und Software Engineering (vgl. Hakiel 1997). So sollten diese auf der einen Seite den langfristigen Effekt der steigenden Arbeitsqualität ihrer Entwickler betrachten und auf der anderen Seite eine messbare Menge an Zielen der Gebrauchstauglichkeit festlegen.

2.2.2. Ebenen der Integration

Ausgehend von dem zuvor diskutierten Bedarf einer Integration und den damit verbundenen Schwierigkeiten, wurde mittlerweile eine Vielzahl von Integrationsansätzen, sowohl aus der Wissenschaft als auch aus der Praxis, formuliert. Betrachtet man die Schwerpunkte dieser Ansätze, so lässt sich nach Nebe (2009) eine Integration auf drei unterschiedlichen Abstraktionsebenen durchführen (siehe Abbildung 2.8).

Standards stellen ein Rahmenwerk auf einer abstrakten, obersten Ebene dar. Mit Hilfe eines solchen Rahmenwerkes wird das Ziel verfolgt, die Konsistenz und Qualität nicht nur innerhalb einer Organisation, sondern auch über deren Grenzen hinweg sicherzustellen und zu wahren. Auf Basis eines Rahmenwerkes können, die darunter liegenden Modelle abgeleitet werden. Standards der Disziplinen des Usability Engineering und des Software Engineering, wie zum Beispiel ISO 9241-210 und ISO/TS 18152 beziehungsweise ISO/IEC 12207 und ISO/IEC 15504 werden in Kapitel 2.3 vorgestellt.

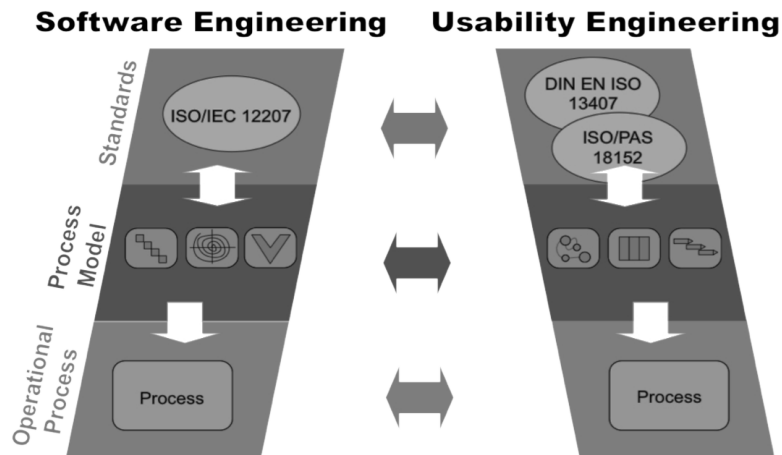


Abbildung 2.8.: Abstraktionsebenen von Integrationsansätzen nach Nebe (2009)

Modelle „beschreiben systematische, planbare und wiederholbare Vorgehensweisen für die Umsetzung und dienen als Vorgabe für die operative Durchführungsebene“ (Nebe 2009). Einige Vorgehensmodelle der Disziplinen wurden exemplarisch in Kapitel 2.1 erläutert.

Im Bereich des Usability Engineering findet man zum Beispiel:

- „Usability Engineering Life Cycle“ (Nielsen 1992)
- „Star Lifecycle“ (Hix & Hartson 1993)
- „Task-Centered User Interface Design“ (Lewis & Rieman 1993)
- „Contextual Design“ (Beyer & Holtzblatt 1997)
- „Usability Engineering Lifecycle“ (Mayhew 1999)
- „Usage Centered Design“ (Constantine & Lockwood 1999)
- „Scenario-based Usability Engineering“ (Rosson & Carroll 2002)
- „Rapid Contextual Design“ (Holtzblatt et al. 2005)
- „Goal-Directed-Design“ (Cooper et al. 2007)

Im Bereich des Software Engineering sind unter anderem folgende Beispiele zu finden:

- „Linear Sequential Model“ (Royce 1970)
- „Adaptive Software Development Process“ (Edmonds 1974)
- „Evolutionäre Softwareentwicklung“ (McCracken & Jackson 1982)
- „Scrum“ (Takeuchi & Nonaka 1982)

- „*Spiralmodell*“ (Boehm 1988)
- „*V-Modell, V-Modell '97*“ (IABG 1995, 1997)
- „*Feature Driven Development*“ (de Luca 1998)
- „*Extreme Programming*“ (Beck 1999)
- „*Crystal Clear*“ (Cockburn 2004)
- „*V-Modell XT*“ (IABG 2009)

Operative Prozesse bilden die konkrete Durchführung von Aktivitäten und die Erstellung von Ergebnissen ab. Dafür werden die Modelle bezüglich der organisatorischen und technischen Rahmenbedingungen der Organisationen beziehungsweise der Projekte angepasst. Dieses Vorgehen bezeichnet man als „*tailoring*“.

Nebe (2009) betont, dass Aspekte aus allen drei Ebenen in einer „vollständigen“ und „umfassenden“ Integration enthalten sein sollten. Diese Aspekte müssen jedoch an den „organisatorischen Kontext“, an die „Komplexität der Organisation“ und an die „Situation“ angepasst werden und dabei die Perspektiven des UE und SE „gleichermaßen“ berücksichtigen.

2.2.3. Existierende Integrationsansätze

Neben den im vorherigen Abschnitt definierten Ebenen von Integrationsansätzen, konnte Nebe (2009) bisherige Ansätze zudem in vier Kategorien einteilen. Diese Kategorien werden in Zusammenhang mit den darin befindlichen Ansätzen im Folgenden näher betrachtet.

1. Konkrete Durchführung und Definition
2. Gemeinsame Spezifikation
3. Definition von Modellen und Prozessen
4. Abstrakte Integrationsansätze

Konkrete Durchführung und Definition Ansätze dieser ersten Kategorie haben nach Nebe (2009) den Fokus auf Aspekten der konkreten Durchführung. Sie definieren Aktivitäten und Artefakte, sowie Verknüpfungen zu bereits bestehenden Aktivitäten des Software Engineering.

Ferre (2003) spezifiziert in seinem Ansatz eine Sammlung von 51 generischen UE Techniken, die von Experten anhand einiger Kriterien ausgewählt wurden. Dabei war entscheidend, wie anpassbar die Techniken an den Softwareentwicklungsprozess sind, die generelle Anwendbarkeit der Techniken, die Kosten für den Einsatz einer Technik und deren generelle Akzeptanz im Gebiet der Mensch-Computer-Interaktion. Es existiert in der Sammlung mindestens eine Technik für jede UE Aktivität, so dass die Gebrauchstauglichkeit des finalen Produktes verbessert wird. Damit die Ergebnisse in Bezug zu den SE Aktivitäten gesetzt werden konnten, mussten diese an die Terminologie und Konzepte des SE angepasst werden.

Constantine et al. (2003) formulieren einen nutzungszentrierten Gestaltungsprozess (*Usage-Centered Design Prozess*), der als robuster und bewährter Prozess auf dem Software Engineering gründen soll. Im Gegensatz zu benutzerzentrierten Gestaltungsprozessen liegt der Fokus hier auf den intendierten Aufgaben der Benutzer und deren Durchführung. Constantine et al. (2003) heben insbesondere die Parallelen zwischen Software Engineering und Usability Engineering hervor, zum Beispiel zwischen *System Use Cases* des SE und *Task Cases* der UE. Auf Grund dieser Zusammenhänge können besonders die drei Modelle des nutzungszentrierten Ansatz (*User Role Model*, *User Task Model* und *Interface Content Model*) in die Prozesse des Software Engineering integriert werden, so dass das Usability Engineering hauptsächlich zu einer verbesserten Benutzerschnittstelle beiträgt.

Schaffer (2004) stellt in seinem Buch „Institutionalization of Usability“ einen Ansatz vor, bei dem der Fokus auf bereits existierenden Produkten liegt. So sollen UE Aktivitäten aus dem Bereich der Evaluation und des Testens auf bestehende Lösungen angewendet werden. Die gewonnen Erkenntnisse fließen dann wieder in den Prozess des Software Engineering ein und führen zu einer Verbesserung des Produktes und einer Integration von UE Aktivitäten.

Gemeinsame Spezifikation In diese Kategorien wurden von Nebe (2009) Ansätze eingeordnet, die auf gemeinsame Spezifikationen oder Artefakte abzielen oder bestehende Artefakte einer Disziplin um Notationen im Sinne der anderen Disziplin erweitern. Durch diese Art der Integration bleibt der Widerstand eher gering und die Kommunikation zwischen den Disziplinen wird gefördert.

Rosson (1999) betont die Abhängigkeiten zwischen der Aufgaben der Benutzer (*User Tasks*) und der dafür entwickelten Software. Durch Charakteristiken der Software und Hardware entstehen Einschränkungen, aber auch Möglichkeiten für die Aufgaben. Da-

her entwickelte Rosson (1999) den Integrationsansatz einer objektorientierten Analyse und Design (*OOAD*) auf Form von *User Interaction Scenarios*. Dies sollen auf der einen Seite den Designern eine Sicht auf die Dynamik der Interaktionsobjekte liefern und auf der anderen Seite den Entwicklern Aufschluss über Angelegenheiten des Gebrauchs und der Software geben.

Seffah et al. (2005a) betrachten in ihrem Ansatz mehrere Rahmenwerke, welche Methoden und Prinzipien der Mensch-Computer-Interaktion mit verschiedenen Methoden des Software Engineering versuchen zu integrieren. Darauf basierend erschufen Seffah et al. (2005a) ein generisches Rahmenwerk, welches eine Integration von Methoden des Usability Engineering in die Praxis von Software Engineering erleichtern und eine „gegenseitige Befruchtung“ der Mensch-Computer-Interaktion und des Software Engineering fördern soll. Dabei argumentieren die Autoren, dass sich vor allem Artefakte, wie *Patterns* und *Use Cases*, für eine Integration eignen.

Juristo et al. (2003) formulieren einen Ansatz, in dem Themen der Gebrauchstauglichkeit nicht erst nach Fertigstellung eines Systems gemessen und bewertet, sondern durch architektonische Muster bereits auf Ebene der Architektur eines Systems verankert werden. Dazu haben Juristo et al. (2003) mehrere Muster des Usability Engineering identifiziert und versuchen diese auf das Software Engineering anzuwenden.

Definition von Modellen und Prozessen Bei Ansätzen, die Nebe (2009) in diese Kategorie einordnete, liegt der Schwerpunkt auf der Definition von Modellen und Prozessen, die als Vorgabe für die Entwicklung gelten. Dabei wird zwischen eigenständigen UE Modellen, wie sie in Abschnitt 2.2.2 - Paragraph *Modelle* aufgelistet sind, und SE Modellen, in denen UE Aktivitäten integriert wurden, unterschieden.

Göransson et al. (2003) analysieren in ihrem Ansatz, in wie weit der Rational Unified Process benutzerzentriert ist. Fazit ihrer Analyse ist, dass RUP kein benutzerzentriertes Vorgehensmodell ist und einen benutzerzentrierten Prozess sogar grundsätzlich verhindert. Dies läge vor allem daran, dass RUP als Modell sehr artefaktlastig ist und somit keine Kollaborationen suggeriert. Deshalb entwickelten Göransson et al. (2003) mit dem „*Usability Design*“ eine weitere Disziplin, die sie in den RUP einfügen. In dieser Disziplin werden benutzerzentrierte Methoden und Werkzeuge formuliert, um Anforderungen zu ermitteln, die zur Gestaltung der Gebrauchstauglichkeit eines Produktes beitragen. Des Weiteren benennen die Autoren mit dem „*Usability Designer*“ eine Rolle, um Belange der Disziplin im Prozess zu repräsentieren, und definieren zwölf Prinzipien

in Form eines Leitfadens für benutzerzentrierte RUPs.

Düchting et al. (2007) betrachten in ihrem Ansatz agile Vorgehensmodelle der Softwareentwicklung, wie Scrum und das Extreme Programming. Diese wurde von den Autoren in Bezug auf die Umsetzung von UE Aktivitäten anhand der DIN EN ISO 13407 überprüft, so dass Empfehlungen in Hinblick einer benutzerzentrierten Umsetzung in der Praxis abgeleitet werden konnten.

Abstrakte Integrationsansätze Ansätze dieser Kategorie sind nach Nebe (2009) abstrakter oder generischer Art. Ohne einen Zusammenhang zu bestimmten SE Modellen verfassen die Autoren dieser Ansätze allgemeine Rahmenbedingungen, welche bei einer Integration von Usability Engineering und Software Engineering zu berücksichtigen sind.

Sousa et al. (2005) beschreiben ein generisches Prozessmodell, welches die Praktiken des Software Engineering und der Mensch-Computer-Interaktion vereint. Dabei verfolgen Sousa et al. (2005) im Wesentlichen drei Ziele. Zum einen sollen Experten beider Disziplinen dabei unterstützt werden ein interaktives System zu entwickeln, welches die Gebrauchstauglichkeit berücksichtigt. Des Weiteren soll die Kommunikation zwischen den beiden Disziplinen verstärkt werden, wodurch die Produktivität im Rahmen einer neuen Arbeitsumgebung verbessert und somit letztendlich die Aspekte der Mensch-Computer-Interaktion als essentieller Teil im Software Engineering verankert werden. Dazu sollen vor allem Aktivitäten und Artefakte der beiden Disziplinen stärker zueinander in Beziehung gesetzt werden. Zudem wird als drittes Ziel eine Grundlage beschrieben, um Benutzerschnittstellen zu erstellen. Dabei sollen Konzepte der Mensch-Computer-Interaktion in den Prozess integriert werden, so dass die Benutzerschnittstellen nicht nur auf den Erfahrungen der Designer beruhen, sondern die Erwartungen der Benutzer widerspiegeln.

Metzker & Reiterer (2002) setzen mit ihrem Ansatz auf der organisatorischen Ebene einer Organisation an und formulieren ein „*Evidence-Based Computer-Aided Usability Engineering Environment (CAUSE)*“. Dabei verwenden sie kein festes Prozessmodell für Usability Engineering, sondern verfolgen das Paradigma einer situationsabhängigen Entscheidungsfindung. Sie argumentieren, dass Organisationen auf einer Reifeskala einen höheren Wert erzielen, wenn die Organisation dazu fähig ist, aus einer Sammlung an *Base Practices* diejenigen auszuwählen, die am Besten auf situationsbedingte Einschränkungen anzuwenden sind. Zu diesen Einschränkungen zählen Metzker & Reiterer (2002) unter anderem verfügbare Ressourcen und Projektcharakteristika. Um ihre Ziele

zu erreichen definieren die Autoren ein Prozess-Metamodell, welches die Auswahl von geeigneten UE Methoden, zum jeweiligen Entwicklungskontext und zur Integration in den Entwicklungsprozess, unterstützt. Des Weiteren beschreiben sie eine semi-formale Notation, um das Wissen aus Erfahrungen angewandeter UE Methoden festzuhalten. Mit Hilfe eines computergestützten Systems, genannt „*ProUSE*“ (*Process centred Usability Engineering Environment*), soll die Auswahl der UE Methoden anhand des Metamodells unterstützt werden. Zuletzt wird ein organisatorisches Konzept formuliert, um die ausgewählten Methoden anzuwenden und die damit gesammelte Wissensbasis auszuwerten.

Wie in diesem Abschnitt beschrieben, existiert bereits eine Vielzahl unterschiedlicher Ansätze zur Integration mit unterschiedlichen Schwerpunkten, die nach Nebe (2009) in vier Kategorien eingeordnet werden können. Der in dieser Arbeit betrachtete Integrationsansatz kann den abstrakten Integrationsansätzen zugeordnet werden, da dieser nicht auf einem bestehenden SE oder UE Modellen aufbaut, sondern generisch anhand von Standards erstellt wurde. Im Gegensatz zum Ansatz von Metzker & Reiterer (2002) werden durch die Basis der Standards keine konkreten Methoden zur Durchführung vorgeschrieben, sondern lediglich Aktivitäten formuliert. Wie diese letztendlich ausgeführt werden, bleibt somit den Organisationen überlassen. Sousa et al. (2005) fokussieren nur wenige Artefakte zur Integration in das Software Engineering. So werden vorwiegend Aufgabenmodelle (*Task Models*), Gebrauchstauglichkeitsanforderungen (*Usability Requirements*) und die Benutzungsschnittstelle (*UI*) betrachtet. Die Analyse des Nutzungskontextes bleibt in diesem Ansatz unberücksichtigt und die Validierung der Ergebnisse wird nicht thematisiert.

Auf Grund der Ausrichtung des Integrationsansatzes dieser Arbeit werden im nächsten Abschnitt Standards des Usability Engineering und Software Engineering vorgestellt.

2.3. Standards

Wie bereits erwähnt, wurde im Vorfeld dieser Arbeit ein ganzheitliches Modell zur Integration erstellt. Dieses gründet im Wesentlichen auf Aktivitäten und Artefakten, die anhand von Standards der Disziplinen ermittelt wurden. Betrachtet man die drei Ebenen einer Integration nach Nebe (2009), so lässt sich das Modell eindeutig auf Ebene der Standards einordnen. Da ein ganzheitliches Modell beide Disziplinen gleichermaßen berücksichtigen sollte, wurde sich bewusst für diese Integrationsebene entschieden. Ein Ansatz auf einer der beiden unteren Ebenen würde bereits von einigen Annahmen ausgehen und hätte so einen eingeschränkten Fokus.

Daher werden in diesem Abschnitt zunächst verwendete Standards erläutert, bevor in Kapitel 3 das erstellte Modell beschrieben und begründet werden kann.

Im deutschsprachigen Raum wird zwischen den Begriffen „Norm“ und „Standard“ unterschieden, während dieser Unterschied im englischsprachigen Raum nicht gemacht wird. Während eine Norm durch breite Beteiligung und durch einen Normierungsprozess verabschiedet wird, wird ein Standard meist nur durch einen geschlossenen Kreis von Unternehmen beschlossen. Im Rahmen der Internationalisierung wird in dieser Arbeit der Begriff des „Standards“ als übergeordneter Begriff verwendet und verstanden nach Voigt (2010) als verbindlich veröffentlichte Aufforderungen oder Regeln in Form von technischen Beschreibungen oder anderen Dokumenten, die für jedermann zugänglich sind und unter Mitarbeit und im Einvernehmen oder mit allgemeiner Zustimmung aller interessierten Kreise erstellt wurden.

Dies entspricht ebenfalls der englischsprachigen Definition der British Standards Institution (BSI 2010):

„Put at its simplest, a standard is an agreed, repeatable way of doing something. It is a published document that contains a technical specification or other precise criteria designed to be used consistently as a rule, guideline, or definition. Standards help to make life simpler and to increase the reliability and the effectiveness of many goods and services we use. Standards are created by bringing together the experience and expertise of all interested parties such as the producers, sellers, buyers, users and regulators of a particular material, product, process or service.“

Standards beruhen auf abgestimmten Ergebnissen von Wissenschaft, Technik und Praxis und sind von einer auf nationaler (*DIN*¹, *BSI*²), regionaler (*CEN*³, *CENELEC*⁴) oder internationaler (*ISO*⁵, *IEC*⁶, *ITU*⁷) Ebene anerkannten Organisation gebilligt worden. (vgl. Voigt 2010)

2.3.1. Nutzen von Standards

Nach Niedziella (2007) hat eine Standardisierung die Vereinheitlichung der Anforderungen an materielle und immaterielle Güter zum Ziel, so dass der Austausch von

¹Deutsches Institut für Normung e.V.

²British Standards Institution

³Comité Européen de Normalisation

⁴Comité Européen de Normalisation Electrotechnique

⁵International Organization for Standardization

⁶International Electrotechnical Commission

⁷International Telecommunication Union

Waren und Dienstleistungen erleichtert wird. Weitere Ziele einer Standardisierung sind Rationalisierung, Kompatibilität, Austauschbarkeit, Gebrauchstauglichkeit, Sicherheit, Umweltschutz, Gesundheit, Verständigung, sinnvolle Ordnung und Information auf einem bestimmten Gebiet, sowie die Qualitätssicherung und Qualitätsverbesserung.

Im Bereich der Softwareentwicklung stellen Standards eine Grundlage für den Prozess der Entwicklung dar oder definieren Anforderungen an das Produkt. Obwohl für eine gute gebrauchstaugliche Software weitaus mehr als Richtlinien und Standards nötig sind, so tragen diese laut Stewart & Travis (2003) jedoch wesentlich zur Konsistenz, einer guten Durchführung bei der Entwicklung und einem gemeinsamen Verständnis bei. Auf Grundlage von Standards werden gewisse Aspekte adressiert, wie zum Beispiel Qualität oder Gebrauchstauglichkeit, welche im Vorfeld gegebenenfalls kaum Aufmerksamkeit erlangt haben.

Die in dieser Arbeit verwendeten Standards lassen sich nach Stewart & Travis (2003) in zwei Kategorien einteilen: *prozessorientierte Standards* und *produktorientierte Standards*. Wie die Namen selbst beschreiben, fokussieren die Standards der ersten Kategorie den Prozess zur Herstellung eines Produktes und die Standards der zweiten Kategorie das eigentliche Produkt.

Erwähnenswert in der Kategorie der prozessorientierten Standards sind vor allem die ISO 9241-210 (2010) und die ISO/IEC 12207 (2002), welche ein Rahmenwerk für die Entwicklung von gebrauchstauglichen Produkten zur Verfügung stellen. Auf der Basis dieser Standards setzen die Standards ISO/TS 18152 (2010) und ISO/IEC 15504 (2006) an und formulieren Anforderungen, um die Reife und Qualität eines Entwicklungsprozesses zu bewerten. Dies geschieht auf Grundlage so genannter *Base Practices*, bei denen es sich um Basisaktivitäten handelt, die auf Grund von gemachten Erfahrungen in der Praxis identifiziert wurden.

In der Kategorie der produktorientierten Standards befinden sich Standards, die Anforderungen und Empfehlungen zum Beispiel an die Gestaltung der Benutzerschnittstelle benennen. So lassen sich die Teile 3-10 und 12-17 der ISO 9241 in diese Kategorie einordnen.

2.3.2. Standards im UE

Besonders in der Disziplin des UE spielen Standards eine wichtige Rolle, da diese dazu beitragen das Thema „Gebrauchstauglichkeit“ im Bereich der Entwicklung interaktiver Systeme zu platzieren. Bereits in den späten 1970'ern wurde mit dem Standard DIN 66-234 die Bedeutung ergonomischer Probleme bei Bildschirmarbeitsplätzen und grafi-

schen Terminals adressiert. Dieser Standard sorgte für internationale Aufmerksamkeit, so dass sich die International Organization for Standardization (*ISO*) diesem Standard annahm und ihn in den Standard ISO 9241-10 überführte, welcher heute unter der Bezeichnung ISO 9241-110 (2006) bekannt ist. (vgl. Stewart & Travis 2003)

Bevan (2001) definiert in seiner Arbeit, anhand der beabsichtigten Ziele, vier Kategorien von Standards (siehe Abbildung 2.9).

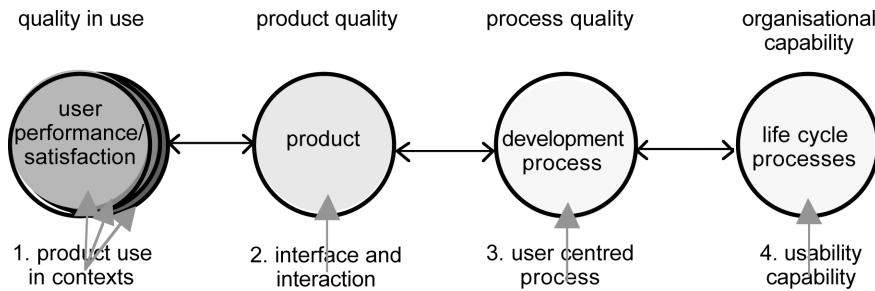


Abbildung 2.9.: Kategorien von UE Standards nach Bevan (2001)

In die erste Kategorie ordnet Bevan (2001) Standards ein, die die Qualität der Benutzung („*quality in use*“), gemäß der in der Definition der Gebrauchstauglichkeit referenzierten Ziele *Effektivität*, *Effizienz* und *Zufriedenstellung* der Benutzer im Nutzungskontext, betreffen. Dazu zählt er vor allem die ISO 9241-11 (1998), welche Anforderungen an die Gebrauchstauglichkeit definiert.

Der Standards ISO 9241 beschreibt eine Richtlinie für die Ergonomie der Mensch-System-Interaktion und besteht aktuell aus 34 Teilen. So gliedert Bevan (2001) die Teile 12-17 und Teil 110 in die zweite Kategorie, da diese Anforderungen an die grafische Benutzerschnittstelle von Bildschirmarbeitsplätzen formulieren und somit auf die Qualität des Produktes abzielen („*product quality*“).

Damit das erstellte Produkt die Ziele der Qualität der Benutzung und der Produktqualität erreichen kann, bedarf es eines benutzerorientierten Gestaltungsprozesses. Ein Rahmenwerk für einen solchen Prozess wird im Standard DIN EN ISO 13407 (1999) beschrieben, welcher mittlerweile in den Standard ISO 9241-210 (2010) übergegangen ist. Die Qualität des Prozesses („*process quality*“) ist Ziel der dritten Kategorie.

Die vierte Kategorie zielt auf organisatorische Fähigkeiten („*organisational capability*“) ab. Diese sind Voraussetzung für die Durchführung eines benutzerorientierten Gestaltungsprozesses. Die ISO/TS 18152 (2010) beinhaltet eine Spezifikation zur Bewertung von Prozessen mit dem Schwerpunkt der Mensch-System Belange und wird von Bevan (2001) in dieser Kategorie genannt.

Im Folgenden werden die wesentlichen Standards des Usability Engineering kurz näher betrachtet.

ISO 9241-210 Die ISO 9241-210 (2010) wurde am 03. März 2010 veröffentlicht und gilt als internationaler Standard für die benutzerorientierte Gestaltung interaktiver Systeme. Der Standard ändert beziehungsweise ersetzt die ehemalige DIN EN ISO 13407 (1999).

Änderungen beziehen sich im Wesentlichen auf die Klarstellung der Rolle der Iteration im gesamten Gestaltungsprozess (nicht nur während der Evaluation) und der Prinzipien einer benutzerzentrierten Gestaltung. Des Weiteren wird die Verwendung von benutzerzentrierten Methoden im Systemlebenszyklus betont und Aktivitäten der Gestaltung erläutert.

Der Standard formuliert ein Rahmenwerk für den Gestaltungsprozess eines interaktiven Produktes im Allgemeinen und ist somit nicht nur auf Software beschränkt, sondern kann auch auf Hardwarekomponenten interaktiver Systeme angewandt werden. Er vervollständigt bestehende Ansätze der Systementwicklung und lässt sich daher in verschiedene Ansätze integrieren, zum Beispiel objektorientierte Entwicklung.

Der Prozess lässt sich in vier Phasen einteilen und betont die iterative Durchführung dieser Phasen (siehe Abbildung 2.10).

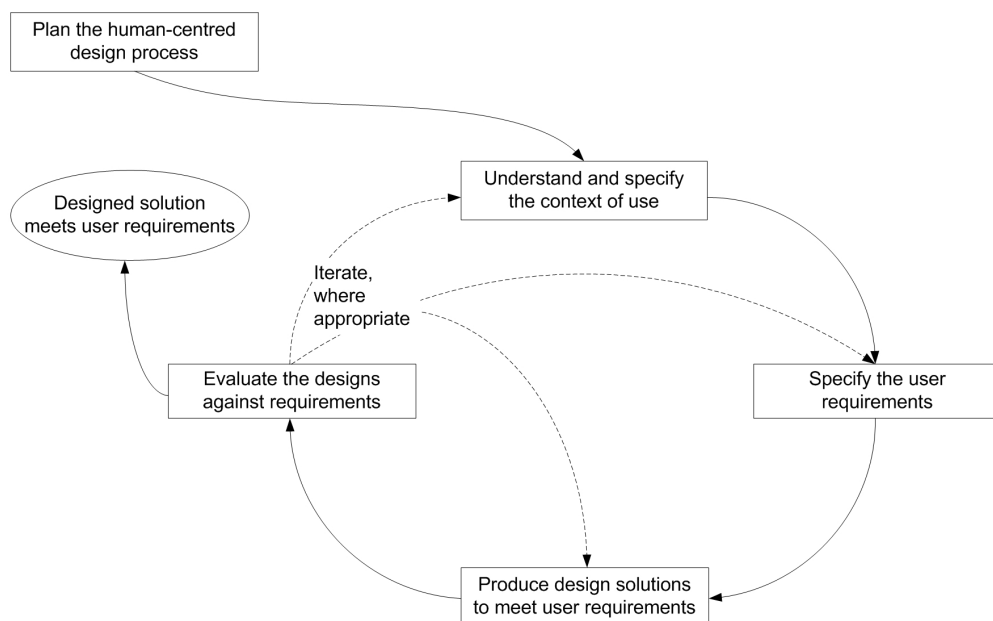


Abbildung 2.10.: Benutzerorientierte Gestaltung interaktiver Systeme (ISO 9241-210)

Nachdem der benutzerzentrierte Gestaltungsprozess geplant wurde, wird zunächst der Nutzungskontext („*Context of Use*“) analysiert und festgelegt. Der Nutzungskontext setzt sich aus Merkmalen der Benutzer, deren Arbeitsaufgaben und -abläufen, sowie der sozialen, kulturellen, physikalischen und organisatorischen Umgebung zusammen. Zudem werden auch Rückmeldungen von Benutzern bei bereits bestehenden Systemen berücksichtigt.

Anhand der zuvor identifizierten Bedürfnisse der Benutzer und dem Nutzungskontext werden in der nächsten Phase die Benutzeranforderungen und organisatorischen Anforderungen („*User Requirements*“) spezifiziert. Dabei werden die Mensch-System Belange dokumentiert und gewichtet, sowie ergonomische und gesetzliche Aspekte, Arbeitsorganisation, Kenntnisse über Benutzerschnittstellen, etc. berücksichtigt. Um spezifizierte Anforderungen und Anforderungen an die Gebrauchstauglichkeit im weiteren Verlauf zu überprüfen, werden messbare Kriterien formuliert.

Die dritte Phase sieht die Entwicklung von Gestaltungslösungen („*Produce Design Solutions*“) vor. Diese werden auf Basis des spezifizierten Nutzungskontextes und der Anforderungen gestaltet und den Benutzern in Form von Prototypen zur Validierung präsentiert. Rückmeldungen der Benutzer werden dann in den Gestaltungslösungen berücksichtigt.

Zum Abschluss jeder Iteration wird die Gestaltungslösung anhand der Anforderungen, dem zu Grunde liegenden Nutzungskontext und den messbaren Kriterien bewertet („*Evaluate the Designs*“). Die Bewertungsergebnisse fließen dann in die darauf folgende Iteration mit ein und nehmen, je nach Stand des Projektes, Einfluss auf eine oder mehrere der zuvor genannten Phasen. Dabei wird eine Gestaltungslösung bewertet, um diese zu verbessern („*formative Evaluation*“), oder die Gestaltungslösung wird auf die zu Beginn formulierten Ziele hin beurteilt („*summative Evaluation*“).

Wichtig für diesen benutzerorientierten Ansatz sind dabei sechs Prinzipien (ISO 9241-210, S. 5):

- „*the design is based upon an explicit understanding of users, tasks and environments*“
- „*users are involved throughout design and development*“
- „*the design is driven and refined by user-centred evaluation*“
- „*the process is iterative*“

- „*the design addresses the whole user experience*“
- „*the design team includes multidisciplinary skills and perspectives*“

Die Analyse des Nutzungskontextes kann als Besonderheit im Gestaltungsprozess genannt werden. So wurde eine eigene Phase für diese Tätigkeit definiert, die grundlegend für alle weiteren Phasen des Prozesses und auch iterativ eingebunden ist. Da eine Iterationsstufe über einen längeren Zeitraum stattfindet, können Veränderungen des Nutzungskontextes wieder in den Prozess mit einfließen. Das Gestaltungsteam erhält somit ein genaues Verständnis darüber, wer die Benutzer sind, wie diese ihre Aufgaben erledigen und wie das Umfeld der Benutzer aussieht und kann basierend darauf die Anforderungen der Benutzer und der Organisation wesentlich detaillierter ermitteln.

Die vier Phasen der ISO 9241-210 sind sehr allgemein gültig formuliert und können daher auch als Rahmenwerk für ein ganzheitliches Modell zur Integration von Usability Engineering und Software Engineering verwendet werden. Die schlichte Struktur des Rahmenwerks ermöglicht so eine weitere Abbildung auf bestehende Prozesse auf organisatorischer Ebene.

Weitere Teile der ISO 9241 Wie bereits erwähnt, besteht die ISO 9241 momentan aus insgesamt 34 Teilen plus drei weiteren, die sich aktuell noch in der Vorbereitung befinden (aufgelistet zum Beispiel im Vorwort der ISO 9241-210). Diese Standards adressieren ergonomische Aspekte, die es bei der Gestaltung von Benutzerschnittstellen und der Interaktion zu berücksichtigen gilt. Daher werden diese Standards auch in der ISO 9241-210 referenziert.

Die ISO 9241-11 „*Guidance on Usability*“ formuliert Leitsätze und Anforderungen an die Gebrauchstauglichkeit und unterstützt somit die Aktivität der Analyse und Definition von Systemanforderungen.

Zur Dialoggestaltung und zur Navigation an der Benutzerschnittstelle tragen die Standards ISO 9241-110 „*Dialogue Principles*“, ISO 9241-14 „*Menu Dialogues*“, ISO 9241-15 „*Command Dialogues*“, ISO 9241-16 „*Direct Manipulation Dialogues*“ und ISO 9241-17 „*Form-filling Dialogues*“ bei, die Aspekte und Prinzipien der Dialoggestaltung beinhalten.

Die Gestaltung und Auswahl von Bildschirmgeräten wird durch die Standards ISO 9241-3 „*Display Requirements*“, ISO 9241-7 „*Requirements for Displays with Reflection*“, ISO 9241-8 „*Requirements for Displayed Colors*“ und ISO 9241-12 „*Presentation*

of Information“ unterstützt.

Die Standards ISO 9241-4 „*Keyboard Requirements*“ und ISO 9241-9 „*Requirements for Nonkeyboard Input Devices*“ tragen zur Auswahl und Gestaltung von Tastaturen und anderen Eingabegeräten bei.

Die Gestaltung des Arbeitsplatzes von Benutzern, die an Bildschirmen arbeiten, wird durch die Standards ISO 9241-5 „*Workstation Layout and Postural Requirements*“ und ISO 9241-6 „*Guidance on the Work Environment*“ adressiert.

Die Unterstützung und das Training von Benutzern findet im Standard ISO 9241-13 „*User Guidance*“ Aufmerksamkeit und die Gestaltung deren Arbeit und Aufgaben wird im Standard ISO 9241-2 „*Guidance on Task Requirements*“ behandelt.

Es lässt sich also erkennen, wie vielfältig die Standards der ISO 9241 sind und wie viele Aspekte es bei der Gestaltung gebrauchstauglicher, interaktiver Systeme zu berücksichtigen gilt. (Stewart & Travis 2003)

ISO/TS 18152 Der Standard ISO/PAS 18152 stammt aus dem Jahre 2003 und besaß zu Beginn dieser Arbeit den Status einer öffentlich zugänglichen Spezifikation („*Publicly Available Specification (PAS)*“). Zur Erläuterung: Sofern die einfache Mehrheit des ISO-Komitees in einer Abstimmung für die Veröffentlichung eines Dokumentes stimmt, so erhält dieses den Status einer PAS. Nach drei Jahren muss dieses Dokument erneut überprüft werden, um dann entweder für drei weitere Jahre den Status PAS zu behalten, zurückgezogen zu werden oder in den Status einer technischen Spezifikation (*TS*) beziehungsweise in den Status eines internationalen Standards (*IS*) übergeht. Den Status PAS darf ein Standards maximal sechs Jahre, also zwei Mal drei Jahre, behalten. Somit wurde die Bezeichnung der ISO/PAS 18152 im Verlauf dieser Arbeit Ende Mai 2010 in die Bezeichnung ISO/TS 18152 umgeschrieben. Dies hatte inhaltlich jedoch keine Auswirkung, sondern basiert lediglich auf der Tatsache, dass die sechs Jahre des Status einer PAS abgelaufen waren. Nun muss der Standard erneut in spätestens sechs Jahren in den Status eines IS übergehen oder zurückgezogen werden. (vgl. ISO 2010a, b)

Die ISO/TS 18152 bietet eine Sichtweise auf die Prozesse eines Systemlebenszyklus mit Schwerpunkt auf Themen, welche auf die Benutzer eines Systems abzielen. Dazu macht der Standard die Inhalte der DIN EN ISO 13407 zugänglich für eine Bewertung. So lässt sich die Fähigkeit einer Organisation hinsichtlich der Durchführung von benutzerorientierten Gestaltungsprozessen bewerten. Die Durchführung einer solchen Bewertung

wird auch als „*Assessment*“ bezeichnet. Dabei sollen folgende vier Ziele verfolgt werden (ISO/TS 18152, S. v):

- „*To provide the means of assessing and mitigating risks arising from human-system issues that will affect usability through the life cycle, both at transition points between life cycle stages and during each stage.*“
- „*To provide a description of human-system processes for use in project planning and for inter-disciplinary communication.*“
- „*As a basis for understanding and cooperation during the tendering process and for human-system capability evaluation to support contract award, either in a stand-alone manner or in conjunction with a software or system capability evaluation.*“
- „*To provide a basis for structured human-system process improvement by supplier, customer or employer organizations.*“

In einem Assessment wird der aktuelle Prozess einer Organisation einem Referenzmodell gegenübergestellt, um zu bewerten, wie sehr der Prozess mit dem Referenzmodell übereinstimmt und ob Mängel im Prozess existieren. Die Reife („*Maturity*“) einer Organisation kann daraufhin anhand eines Bewertungsmodells eingestuft werden. Die ISO/TS 18152 definiert dabei kein eigenes Bewertungsmodell, sondern ist konform mit der ISO/IEC 15504. So lässt sich der gesamte Entwicklungsprozess anhand der beiden Standards bewerten.

Das Referenzmodell der ISO/TS 18152 umfasst die vier Prozesskategorien:

- HS.1 Life cycle involvement
- HS.2 Integrate human factors
- HS.3 Human-centred design
- HS.4 Human resources

Jede Prozesskategorie enthält so genannte „*Base Practices*“, die häufig auch als „*Best Practices*“ oder „*Good Practices*“ bezeichnet werden. Eine *Base Practice* beschreibt eine Aktivität, die zur Erstellung eines *Outputs* (zum Beispiel eines Artefakts) in einem Prozess beiträgt oder die Fähigkeit eines Prozess erweitert (ISO/IEC 15504:2006).

Prozesskategorie HS.1 beinhaltet Base Practices, welche die Interessen und Bedürfnisse der Stakeholder in den gesamten Lebenszyklus eines Produktes, von der Konzeption über die Entwicklung bis zur Produktabkündigung, verankern.

Auf eine zufriedenstellende Eingliederung des benutzerzentrierten Ansatzes in die verschiedenen Bereiche der Organisation (Geschäftsstrategie, Qualitätsmanagement, Management, etc.) zielt die Prozesskategorie HS.2 ab.

Die Prozesskategorie HS.3 enthält Base Practices eines benutzerorientierten Gestaltungsprozesses und basiert auf der DIN EN ISO 13407. Die vier Subprozesskategorien adressieren dabei die vier Phasen des Standards.

Prozesskategorie HS.4 formuliert Base Practices zur Organisation der menschlichen Ressourcen, also die Auswahl eines geeigneten Personals mit Fähigkeiten zur Durchführung von benutzerorientierten Gestaltungsaktivitäten.

Eine vollständige Auflistung der Base Practices befindet sich an Anhang B.1.

Die ISO/TS 18152 stellt somit eine ausführliche Bewertungsgrundlage dar, mit der die Reife des benutzerorientierten Prozesses einer Organisation bewertet beziehungsweise verbessert werden können.

2.3.3. Standards im SE

Auch im Software Engineering spielen Standards eine wichtige Rolle. Sie definieren ein Rahmenwerk für die Entwicklung von Software und ermöglichen die Überprüfung des Entwicklungsprozesses, sowie erstellter Artefakte. Zwei sehr weit verbreitete Standards im Software Engineering sind ISO/IEC 12207 (2002) und ISO/IEC 15504 (2006), die im Folgenden näher betrachtet werden.

ISO/IEC 12207 Der Standard ISO/IEC 12207 wurde im Jahre 1995 durch die ISO und die IEC veröffentlicht. Im Jahre 2002 wurde der Standard durch die Ergänzungsversion Amendment 1 erweitert, welche in dieser Arbeit betrachtet wird. Mittlerweile ist eine neue Version des Standards im September 2008 erschienen, die im Bestand der Hochschule nicht zur Verfügung stand.

Wesentliche Prozesse des Lebenszyklus einer Softwareentwicklung (Akquise, Bereitstellung, Entwicklung, Betrieb, Wartung und Pflege) werden in diesem Standard adressiert (siehe Abbildung 2.11), so dass der Standard als Rahmenprozess zur Entwicklung und dem Management einer Software genutzt werden kann.

Diese Hauptprozesse („*Primary Life Cycle Processes*“) werden durch weitere Prozesse unterstützt. „*Supporting Life Cycle Processes*“ zielen dabei auf die Dokumentation des Gesamtprozesses und die gewonnen Erkenntnisse beziehungsweise Artefakte ab. Des Weiteren sind Prozesse zur Gewährleistung der Qualität („*Quality Assurance*“), zur formalen und inhaltlichen Überprüfung des Systems („*Verification, Validation*“), zur

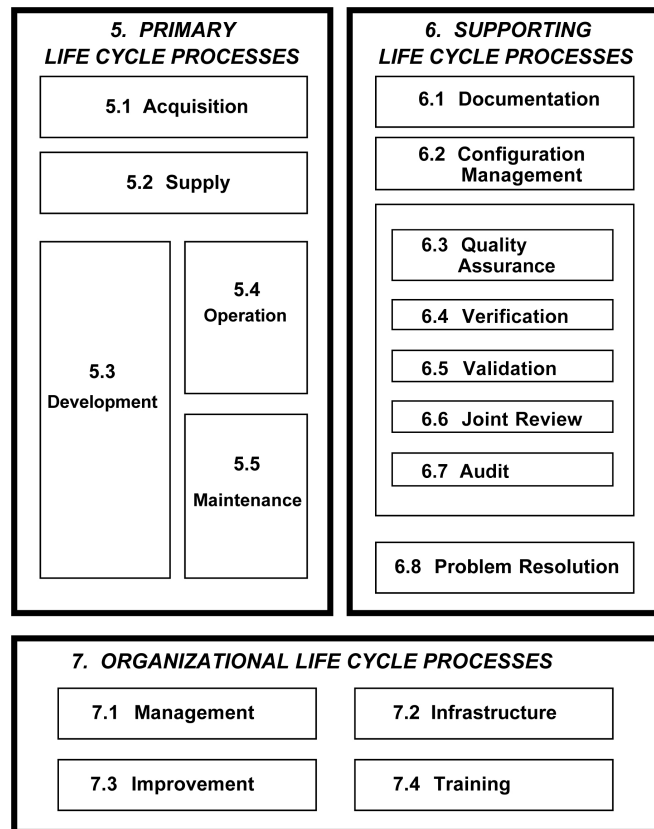


Abbildung 2.11.: Struktur der ISO/IEC 12207

Abstimmung mit dem Kunden („*Joint Review*“), Audits und Prozesse zur Problemlösung („*Problem Resolution*“) beinhaltet.

Organisatorische Prozesse adressieren das Management und die Verbesserung der Prozesse einer Organisation und die Erstellung einer geeigneten Infrastruktur von Entwicklungsumgebungen und Personal.

Jeder Prozess setzt sich dabei aus Aktivitäten zusammen, die wiederum aus eine Reihe von Aufgaben bestehen. Wie diese Aufgaben auszusehen haben, wird jedoch nicht in dem Standard beschrieben, sondern erst im Vorgehensmodell der jeweiligen Organisation festgelegt. Diese entscheidet auch über die durchzuführenden Methoden.

Der Standard formuliert also ausschließlich einen Rahmen für einen Prozess, der von einer Organisation auf deren Bedürfnisse angepasst werden kann.

ISO/IEC 15504 Der Standard ISO/IEC 15504 hat die Bewertung der Reife von Softwareprozessen einer Organisation zum Ziel. Er wurde im Jahre 1998 als Technischer Report (*TR*) veröffentlicht und ist ebenfalls unter dem Begriff SPICE („*Software Pro-*

cess Improvement and Capability dEtermination“) bekannt, dem Ursprungsprojekt des Software Engineering Institute (SEI) der Carnegie Mellon University in Pittsburgh, USA. Der Standard ist im Jahre 2004 veröffentlicht worden und besteht seit 2006 aus mittlerweile fünf Teilen.

Teil 1 enthält eine allgemeine Einführung in das Konzept einer Prozessbewertung und definiert damit verbundene Begriffe.

Teil 2 stellt das Bewertungsmodell dar (siehe Tabelle 2.1) und formuliert formale Anforderungen zur Durchführung einer Prozessbewertung, so dass die Konsistenz und eine Wiederholbarkeit gewährleistet sind. Somit kann der Beweis zur Einschätzung der Reife eines Prozesses erbracht und verifiziert werden.

Teil 3 beinhaltet Anforderungen zur konkreten Durchführung einer Bewertung und beschreibt die Reifegrade.

Teil 4 identifiziert die Bewertung eines Prozesses als eine Aktivität, die entweder im Rahmen einer Prozessverbesserung oder zur generellen Bestimmung der Reife des Prozesses durchgeführt werden kann. Das Ziel einer Prozessverbesserung besteht darin, die Effektivität und Effizienz einer Organisation zu verbessern. Die Bestimmung der Reife eines Prozesses dient der Identifikation von Stärken, Schwächen und Risiken innerhalb eines Prozesses ausgerichtet an den Zielen der Geschäftsstrategie.

Teil 5 liefert ein exemplarisches Prozessbewertungsmodell, welches auf dem in der ISO/IEC 12207 spezifizierten Referenzmodell basiert. So kann der Prozess einer Organisation mit dem Prozessbewertungsmodell abgeglichen und daraufhin die Reife des Prozesses bestimmt werden.

Reifestufe (Level)		Beschreibung
0	Incomplete	The process is not implemented, or fails to achieve its process purpose.
1	Performed	The implemented process achieves its process purpose.
2	Managed	The process is now implemented in a managed fashion (planned, monitored and adjusted) and its work products are appropriately established, controlled and maintained.
3	Established	The process is now implemented using a defined process that is capable of achieving its process outcomes.
4	Predictable	The process now operates within defined limits to achieve its process outcomes.

Weiter auf der nächsten Seite

Reifestufe (Level)		Beschreibung
5	Optimizing	The process is continuously improved to meet relevant current and projected business goals.

Tabelle 2.1.: Reifestufen nach Teil 2 ISO/IEC 15504

Die ISO/IEC 15504 stellt, genau wie die ISO/TS 18152, eine umfangreiche Bewertungsgrundlage dar, mit dem Ziel die Reife der Prozesse eines Softwarelebenszyklus zu analysieren, zu bewerten und zu optimieren.

2.3.4. Folgerungen für diese Arbeit

Wie bereits erwähnt, zielen die hier vorgestellten Standards auf eine Vereinheitlichung, eine Optimierung und die Qualität der Softwarelebenszyklusprozesse in Organisationen ab.

Der Standard ISO/IEC 12207 liefert dazu ein abstraktes Rahmenwerk, welches sich in Prozesse, Aktivitäten und Aufgaben gliedert. Es ist abstrakt, da keine Aussage darüber getroffen wird, wie diese Aufgaben aussehen und welche Methoden anzuwenden sind. Der Standard ISO 9241-210 formuliert ebenfalls ein abstraktes Rahmenwerk, welches sich speziell auf den benutzerorientierten Gestaltungsprozess interaktiver Produkte bezieht. Dieses ist unterteilt in vier Phasen und betont die iterative Durchführung der darin enthaltenen Aktivitäten.

Da diese Standards keine konkreten Aufgaben formulieren oder Methoden vorschreiben, können sie je nach Softwareentwicklungsprojekt entsprechend angepasst werden. Eine Integration des Usability Engineering und Software Engineering auf Ebene der Standards (vgl. Nebe 2009) würde also dazu führen, dass sich das ganzheitliche Rahmenwerk nach einer Integration wieder auf die Prozesse der Organisationen abbilden ließe. Eine Integration auf Ebene von Modellen oder des operativen Prozesses würde dies einschränken. Ausgehend von einer Integration auf Ebene der Standards, könnte dies sogar in Vorgehensmodelle adaptiert werden, so dass eine Konsistenz und Qualität nicht nur innerhalb einer Organisation, sondern auch über deren Grenzen hinweg sichergestellt werden kann.

Zudem eignen sich die in den beiden Standards formulierten Aktivitäten, um nach Anknüpfungspunkten der beiden Disziplinen zu suchen, indem sämtliche Aktivitäten in einem ganzheitlichen Modell abgebildet werden. Als gemeinsamer Rahmen dafür eignen sich die vier Phasen der ISO 9241-210, da diese sehr allgemein formuliert sind und in der Praxis bereits aus dem Software Engineering anerkannt werden (vgl. Seffah et al. 2005b).

Die Standards ISO/TS 18152 und ISO/IEC 15504 bieten eine umfassende Bewertungsgrundlage. So können die identifizierten Aktivitäten der Standards ISO 9241-210 und

ISO/IEC 12207 anhand dieser Bewertungsgrundlage in Bezug auf ihre Vollständigkeit und intrinsische Validität hin überprüft werden. Intrinsisch, da der Standard ISO/TS 18152 auf der ISO 9241-210 und der Standard ISO/IEC 15504 auf der ISO/IEC 12207 basiert beziehungsweise diese als Rahmenmodell referenziert.

Anhand der von Nebe (2009) definierten Kategorien von Integrationsansätzen, ließe sich dieser Ansatz somit in die vierte Kategorie der abstrakten Integrationsansätze einordnen, da dieses Modell keinen Bezug zu konkreten UE Modellen oder SE Modellen besitzt und von abstrakter und generischer Art wäre.

2.4. Zusammenfassung

In diesem Kapitel wurden Grundlagen erläutert, die für das generelle Verständnis dieser Arbeit notwendig sind. Dazu wurden zunächst die am Entwicklungsprozess beteiligten Disziplinen des Usability Engineering und Software Engineering anhand deren Entstehung, deren Zielen und einigen Vorgehensmodellen betrachtet.

Auf Grund der Tatsache, dass die „Gebrauchstauglichkeit“ von Produkten mittlerweile als wesentliche Qualitätseigenschaft in der Softwareentwicklung angesehen wird und auch die Organisationen den Mehrwert von gebrauchstauglichen Produkten erkannt haben, wird der Wissenschaft und Praxis momentan dem Wunsch einer Integration dieser beiden Disziplinen nachgegangen. So konnte in diesem Kapitel der Bedarf einer Integration verdeutlicht und verschiedene Ebenen der Integration (*Standards, Modelle, operativer Prozess*) nach Nebe (2009) betrachtet werden. Des Weiteren wurden einige existierende Integrationsansätze vorgestellt, die nach Nebe (2009) in vier Kategorien (*Konkrete Durchführung und Definition, Gemeinsame Spezifikation, Definition von Modellen und Prozessen, Abstrakte Integrationsansätze*) eingeordnet werden konnten.

Da Standards eine wichtige Rolle in den Disziplinen spielen und zugleich die oberste Integrationsebene darstellen, wurden diese ebenfalls in diesem Kapitel beschrieben. So konnten Folgerungen für diese Arbeit getroffen werden, die ausschlaggebend für das folgende Kapitel sind, in dem ein Modell zur Integration der beiden Disziplinen vorgestellt wird.

3. Integrationsmodell

Im Vorfeld dieser Arbeit wurde bereits an einem Modell zur Integration gearbeitet, welches das Usability Engineering und das Software Engineering auf Ebene von Standards integriert. Dabei wurden anhand von Aussagen in der DIN EN ISO 13407, für das Usability Engineering, und in der ISO/IEC 12207, für das Software Engineering, eigenständig Aktivitäten und Artefakte identifiziert und in einem ganzheitlichen Integrationsmodell abgebildet (siehe Abschnitt 3.1). Daher wird in diesem Kapitel zunächst die Struktur des Modells und die auf dessen Basis gewonnenen Erkenntnisse erläutert. Im Rahmen dieser Arbeit sollte das Modell ergänzt und validiert werden. Da, wie im vorherigen Kapitel bereits erwähnt, der UE-Standard DIN EN ISO 13407 im März 2010 vom Standard ISO 9241-210 abgelöst wurde, fand zunächst eine Anpassung des existierenden Modells statt. Dazu wurden erneut, anhand desselben Vorgehens, auf Basis der Aussagen im Standard Aktivitäten und Artefakte ermittelt, um die aus der DIN EN ISO 13407 mit den neu identifizierten zu ersetzen. Des Weiteren wird der Abgleich des Modells auf den Standard ISO/TS 18152 dargestellt, da dieser für die darauf folgende Evaluation, die Kernaktivität dieser Arbeit, notwendig ist. Dabei wurden die im Standard formulierten Base Practices und Work Products mit denen im Modell ermittelten Aktivitäten und Artefakten verglichen und ergänzt. Um die Konstruktvalidität des entwickelten Modells sicherzustellen, wird dieses mit Konformitäts- und Rahmenanforderungen an einen Softwareentwicklungsprozess abgeglichen. Diese Konformitäts- und Rahmenanforderungen beruhen auf der Dissertation von Nebe (2009), der diese erarbeitet und formuliert hat, sowie durch Experten validieren ließ. Dafür werden die Anforderungen auf die Aktivitäten, Artefakte und das Integrationsmodell als Ganzes abgebildet, um gegebenenfalls ergänzende Aktivitäten und Artefakte zu identifizieren, sowie bestehende Aktivitäten, Artefakte und Wechselwirkungen im Modell zu validieren. Anhand der Resultate des Abgleiches entsteht ein angepasstes Modell, welches abschließend in diesem Kapitel betrachtet wird.

3.1. Ausgangsbasis

Diese Arbeit beschäftigt sich mit einem ganzheitlichen Modell der Integration von Usability Engineering und Software Engineering. In Kapitel 2.2 wurde Integration als Bil-

dung einer Einheit definiert, unter der gleichmäßigen Berücksichtigung von Interessen und Zielen beider Disziplinen.

Nach Nebe (2009) ist das Ziel von Usability Engineering, dass „[...] durch den Prozess der Erstellung sichergestellt werden [kann], dass die Lösung eine angemessene Qualität aufweist“. Qualität bezieht sich damit auf den gesamten Prozess und nicht nur auf das entwickelte Produkt. Daher sollten nicht einzelne, sondern sämtliche Aktivitäten des Usability Engineering mit denen des Software Engineering integriert werden, damit durch das Usability Engineering Einfluss auf den gesamten Prozess besteht und nicht nur auf Teilprozesse.

Insbesondere größere Unternehmen (>100 Mitarbeiter) bestehen meist aus einer umfangreichen Hierarchie, in der gewisse Richtlinien, beispielsweise eine spezielle Dokumentation von Ergebnissen oder Entscheidungsgenehmigungen, eingehalten werden müssen. Des Weiteren werden häufig spezielle, auf das Unternehmen angepasste Vorgehensmodellen, sowie unternehmensspezifische Termini verwendet. Damit die Komplexität eines Unternehmens und deren organisatorischer Kontext das Integrationsmodell nicht einschränkt, wurde dieses auf der von Nebe (2009) definierten Ebene der Standards entwickelt (vergleiche Kapitel 2.2.2). Standards können so über die Grenzen eines Unternehmens hinweg angewendet werden. Zudem sind in den Standards keine konkreten Aufgaben angegeben oder Methoden definiert, so dass sich das ganzheitliche Rahmenwerk an den jeweiligen operativen Prozess anpassen lässt.

Ziel der Integration auf Ebene der Standards soll sein, Anknüpfungspunkte zwischen dem Usability Engineering und dem Software Engineering zu ermitteln.

3.1.1. Auswahl der Standards

Als anerkannter Konsens der Experten aus Wissenschaft und Praxis wurde für das Usability Engineering zunächst der Standard DIN EN ISO 13407⁸ (1999) und für das Software Engineering der Standard ISO/IEC 12207 (2002) ausgewählt.

Beide Standards repräsentieren jeweils ein abstraktes Rahmenwerk mit Phasen, Aktivitäten und Artefakten, anhand derer konkrete Vorgehensmodelle (beispielsweise das V-Modell, der Rational Unified Process, der Usability Engineering Lifecycle oder das Scenario-based Development) abgeleitet und operativ durchgeführt werden können. Vorgestellt wurden diese Standards bereits in Kapitel 2.3.2 und 2.3.3. Zusammenfassend lassen sich die Standards anhand ihrer Beschreibungen wie folgt formulieren:

⁸ANMERKUNG: Zum Zeitpunkt der Vorarbeiten zur Masterthesis galt die DIN EN ISO 13407 (1999) noch als aktuell und wurde erst im weiteren Verlauf durch die ISO 9241-210 (2010) ersetzt. Der Umgang mit dieser Neuerscheinung wird in Kapitel 3.2 behandelt.

DIN EN ISO 13407 „*This International Standard provides guidance on human-centred design activities throughout the life cycle of computer-based interactive systems. It is aimed at those managing design processes and provides guidance on sources of information and standards relevant to the human-centred approach.*“

ISO/IEC 12207 „*This International Standard establishes a common framework for software life cycle processes, with well-defined terminology, that can be referenced by the software industry. It contains processes, activities, and tasks that are to be applied during the acquisition of a system that contains software, a stand-alone software product, and software service and during the supply, development, operation, and maintenance of software products.*“

Während in der DIN EN ISO 13407 Aktivitäten in Bezug auf die Gestaltung betrachtet werden, wird in der ISO/IEC 12207 der gesamte Lebenszyklus einer Software festgelegt, inklusive zum Beispiel dem Management, der Auslieferung oder der Wartung. Da Anknüpfungspunkte der Disziplinen somit vorwiegend in der Entwicklung von Software stattfinden können, wurde der Schwerpunkt des Integrationsmodells auf diesen Bereich gelegt. Dem zu Folge wurde hauptsächlich der so genannte „*Development Process*“, Abschnitt 5.3 der ISO/IEC 12207, im Modell berücksichtigt. Anhand der relevanten Seitenzahlen der Standards in Bezug zu den gesamten Seitenzahlen, kann die Vollständigkeit der Betrachtung mit den in Tabelle 3.1 angegebenen Prozentsätzen beziffert werden.

DIN EN ISO 13407:	100%
ISO/IEC 12207:	54%

Tabelle 3.1.: Prozentuale Verwendung der Standards

3.1.2. Ermitteln der Aktivitäten und Artefakte

Anhand der Standards wurden im ersten Schritt Aktivitäten und Artefakte extrahiert und benannt. Dazu wurden textuelle Aussagen zu Tätigkeiten oder Ergebnissen innerhalb der Standards ermittelt und sprachlich überführt in Aktivitäten beziehungsweise Artefakte.

So wurde zum Beispiel die Aussage

„*The context-of-use description shall include [...] c) The goals and tasks of the users*“

überführt in die Aktivität

„Identifizieren der Aufgaben der Benutzer und damit verbundene Ziele“

Sämtliche identifizierten Aktivitäten und Artefakte wurden auf Notizzetteln vermerkt (siehe Abbildung 3.1) und auf einer Pinnwand zusammengetragen, um so einen generellen Überblick zu erhalten. Ein Notizzettel enthielt die Beschreibung der Aktivität beziehungsweise des Artefakts, sowie eventuelle Zusatzinformationen und eine Referenz auf den Textblock im Standard. Zudem wurde farblich unterschieden zwischen dem Usability Engineering und dem Software Engineering.

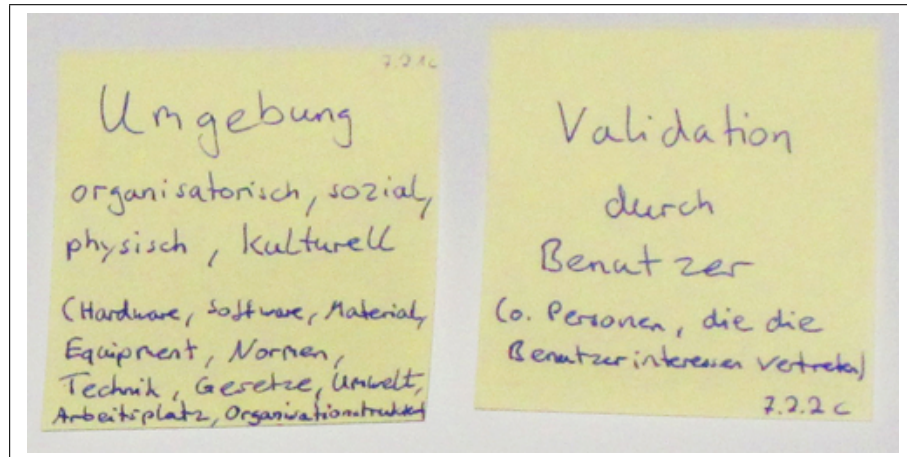


Abbildung 3.1.: Extrahierte Aktivität auf einem Notizzettel

Für das Usability Engineering konnten somit, anhand der DIN EN ISO 13407, zunächst 28 Aktivitäten und 26 Artefakte ermittelt werden. Im Software Engineering, auf Basis der ISO/IEC 12207, belief sich die Anzahl der Aktivitäten auf 65 und die Anzahl der Artefakte auf 37.

Um Gemeinsamkeiten und somit potentielle Anknüpfungspunkte der Disziplinen auf Grundlage der Aktivitäten und Artefakte aufzuzeigen, wurde in einem zweiten Schritt die Technik des Affinitätsdiagrammes verwendet.

Diese Technik wurde als eine der „seven quality processes“ aus Japan vorgestellt und ist durch Kawakita (1982) auch bekannt als „KJ Method“. Mittels dieser Technik soll die Organisation der einzelnen Notizzettel unterstützt werden, die während der Interpretationsphase gesammelt wurden. Nach Beyer & Holtzblatt (1997) werden zunächst Oberthemen gebildet, denen sich die einzelnen Notizzettel zuordnen lassen. So soll eine hierarchische Struktur entstehen, in der Punkte mit ähnlicher Ausrichtung oder gemeinsamen Zielen gruppiert werden.

Der Technik zu Folge wurden so zunächst Oberthemen für die Gruppierung gebildet. Dabei wurden die Phasen der DIN EN ISO 13407 für Oberthemen als sinnvoll erachtet. Diese sind allgemein formuliert und generischer Natur. Zudem sind diese nach Seffah et al. (2005b) in der Praxis auch durch das Software Engineering anerkannt, da sie sich auf die generischen Prinzipien von „Analyse & Design“, „Entwicklung“ und „Evaluation“ abbilden lassen. So konnten folgende sechs Oberthemen festgelegt werden (siehe Abbildung 3.2):

- Organisatorische Vorbereitungen
- Verstehen und Beschreiben des Nutzungskontextes
- Festlegen von Nutzungsanforderungen
- Entwerfen von Gestaltungslösungen
- Beurteilen von Gestaltungslösungen
- System erfüllt festgelegte Anforderungen



Abbildung 3.2.: Grundmodell der sechs Oberthemen (Phasen)

Nach der Festlegung dieser Oberthemen wurden die Notizzettel neu strukturiert und auf der Pinnwand angeordnet. Zudem wurden Aktivitäten und Artefakte beider Disziplinen räumlich nebeneinander platziert, soweit anhand der textuellen Beschreibungen in den Standards eine Gemeinsamkeit vermutet wurde. Ein Beispiel für eine vermutete Gemeinsamkeit besteht so zwischen den Aussagen

„Deriving user requirements“ (DIN EN ISO 13407)

und

„User operation and execution requirements“ (ISO/IEC 12207)

Das so entstandene Ergebnis lässt sich in Abbildung 3.3 betrachten.



Abbildung 3.3.: Affinitätsdiagramm der Aktivitäten

3.1.3. Digitalisierung der Ergebnisse

Im nächsten Schritt wurden die auf der Pinnwand notierten Ergebnisse digitalisiert. Zur Diskussion standen dabei verschiedene Werkzeuge:

Eclipse Process Framework (EPF) ⁹ Das Eclipse Process Framework basiert auf der Eclipse Entwicklungsumgebung und ist als OpenSource frei verfügbar. Es bietet ein erweiterbares Rahmenwerk und exemplarische Werkzeuge zur Prozessentwicklung im Bereich der Softwareentwicklung und des Softwaremanagements. [OpenSource]

IBM Rational Method Composer ¹⁰ Der IBM Rational Method Composer ist eine Erweiterung des Eclipse Process Framework und stellt eine flexible Prozessplattform, basierend auf dem Rational Unified Process, dar. Es bietet diverse Werkzeuge und Prozessbibliotheken zur Unterstützung von Organisationen bei der

⁹Eclipse Foundation (Hrsg.): <http://www.eclipse.org/epf/>, Sichtung: 05.03.2010

¹⁰IBM (Hrsg.): <http://www-142.ibm.com/software/products/de/de/rmc>, Sichtung: 05.03.2010

Implementierung effektiver Prozesse im Bereich von Software- und IT-Projekten.
[Lizenz, ca. 500 €]

Microsoft® Office Visio ¹¹ Microsoft® Office Visio ist eine Software zur Visualisierung und Analyse komplexer Informationen, Systeme und Prozesse. So lassen sich mittels einer Palette an Vorlagen unter anderem Flussdiagramme für Geschäftsprozesse, Workflowdiagramme oder Organigramme erstellen. [Lizenz, ca. 330 €]

OmniGraffle ¹² Ähnlich wie Visio ist auch OmniGraffle eine Software zur Visualisierung. So ermöglicht es unter anderem die Erstellung von Prozessdiagrammen und stellt Strukturen und Abläufe bildlich dar. [Lizenz, ca. 100 €]

Microsoft® StickySorter ¹³ StickySorter von den Microsoft® Office Labs ist eine Software zum Anlegen und Sortieren digitaler Notizzettel. Inspiriert von realen Problemen bei ihrer täglichen Arbeit, wurde das Programm von zwei Mitarbeitern entwickelt und kostenfrei für den Zweck der Rückmeldungen von Benutzern verfügbar gemacht. Es unterstützt eine virtuelle Kollaboration und sichert so zum Beispiel die Ergebnisse eines Brainstormings. [Freeware]

Nach Betrachtung der Alternativen ließ sich feststellen, dass sowohl das Eclipse Process Framework als auch der IBM Rational Method Composer zwei mächtige Werkzeuge darstellen und auch Iterationen theoretisch ermöglichen. Jedoch werden Iterationen lediglich durch ein Symbol verdeutlicht, so dass die gesamte Struktur eher einem sequenziellen Modell gleicht. Zudem war die Modellierung mit diesen Werkzeugen derart komplex, dass einfache Abbildungsmöglichkeiten von Ähnlichkeiten oder Gleichheiten von Aktivitäten nicht möglich waren. Des Weiteren mussten Informationen an mehreren Positionen redundant angegeben werden, damit die Ausgabe dieser Werkzeuge die Informationen auch an den gewünschten Positionen darstellt.

Der StickySorter hingegen war ein sehr einfaches Werkzeug, dass es ermöglichte die Notizzettel der analogen Version auf dem Whiteboard originalgetreu abzubilden. Jedoch beinhaltete das Werkzeug einen Fehler, so dass die Anordnung der Notizzettel nach dem Speichern und erneuten Öffnen der Datei nicht mehr der Version vor dem Speichern entsprach und die Notizzettel wieder willkürlich im Dokument verteilt waren.

¹¹Microsoft® Corporation (Hrsg.): <http://office.microsoft.com/de-de/visio/default.aspx>, Sichtung: 05.03.2010

¹²The Omni Group (Hrsg.): <http://www.omnigroup.com/products/OmniGraffle/>, Sichtung: 05.03.2010

¹³Microsoft® Office Labs (Hrsg.): <http://www.officelabs.com/projects/stickysorter/Pages/default.aspx>, Sichtung: 05.03.2010

Microsoft® Office Visio und OmniGraffle sind ebenfalls zwei mächtige Werkzeuge, die jedoch auf eine visuelle Darstellung abzielen und so auch Iterationen, räumliche Nähe und Verknüpfungen der einzelnen Aktivitäten ermöglichen. Zudem unterstützen beide Werkzeuge den Export des Dokumentes in verschiedene Formate. So können die Dokumente neben dem Export als Grafik oder PDF auch in eine XML Struktur ausgegeben werden, so dass eine theoretische Weiterverarbeitung der Daten möglich wäre. Folglich wurde die Digitalisierung letztendlich mit OmniGraffle realisiert, welches auch eine Schnittstelle zu Microsoft® Office Visio bietet, so dass auch dieses Werkzeug geeignet schien.

Analog zu den Notizzetteln auf der Pinnwand wurden diese originalgetreu digitalisiert und angeordnet. So werden Aktivitäten durch ein Viereck und Artefakte in Form eines Fünfecks dargestellt (siehe Abbildung 3.4).

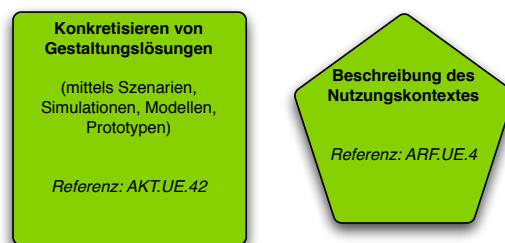


Abbildung 3.4.: Beispiel für ein/e digital repräsentierte/s Aktivität (links) und Artefakt (rechts)

Jedes Objekt enthält dabei vier Informationen. Neben dem Namen der Aktivität oder des Artefakts, werden auch eventuelle Zusatzinformationen, die zum Verständnis beitragen, sowie die Referenz auf den Textblock im Standard erfasst. Die vierte Information über die jeweilige Disziplin, welche diese Aktivität ausführt oder dieses Artefakt nutzt beziehungsweise erzeugt, wird durch die Hintergrundfarbe repräsentiert.

Bei der anfänglichen Analyse der Standards wurde deutlich, dass gewisse Abhängigkeiten zwischen Aktivitäten und Artefakten existieren. So scheint eine Art Lebenszyklus von Artefakten zu existieren. Dem zu Folge handelt es sich bei Artefakten nicht nur um Ergebnisse, die in Aktivitäten erzielt werden. Artefakte dienen ebenfalls als Ausgangsposition für weitere Aktivitäten und werden gegebenenfalls sogar benötigt, um diese überhaupt durchführen zu können. Dies führt zudem dazu, dass bestimmte Artefakte andere Artefakte voraussetzen.

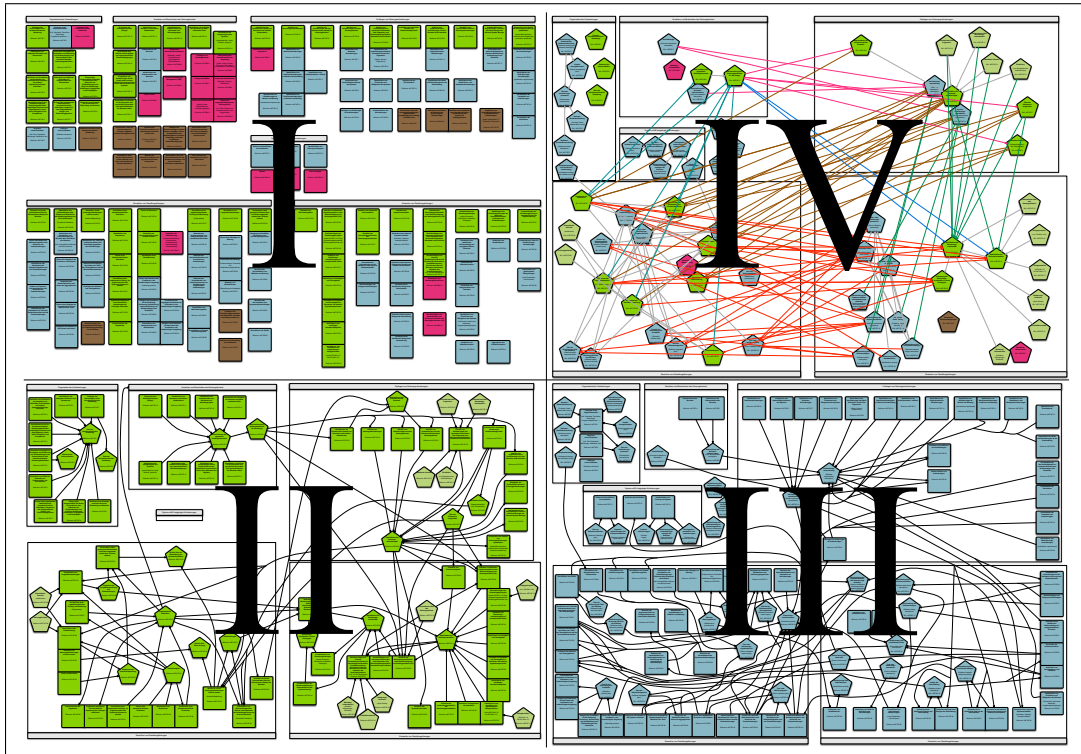


Abbildung 3.5.: Gesamtansicht des Integrationsmodells

Auf Grund der so entstandenen Komplexität wurden vier Sichten auf das erstellte Modell von Aktivitäten und Artefakten (siehe Abbildung 3.5) erzeugt, die im Folgenden als vier Teilmodelle bezeichnet, erläutert und in exemplarischen Ausschnitten dargestellt werden (siehe Abbildungen 3.6 bis 3.9). Die vollständigen finalen Teilmodelle können in Anhang G betrachtet werden. In Ergänzung zu den visuellen Modellen wurden die Informationen zu Aktivitäten, Artefakten, deren Lebenszyklus, sowie den vermuteten Gemeinsamkeiten, inklusive den Referenzen zu den Standards, in Tabellen gelistet. Diese befinden sich in Anhang A, C und D.

I) Teilmodell der Aktivitäten Um Gemeinsamkeiten oder Ähnlichkeiten zwischen Aktivitäten der Disziplinen zu identifizieren, wurde anhand des Modells eine Sicht mit sämtlichen Aktivitäten erzeugt (siehe Abbildung 3.6). In diesem Teilmodell wurden alle identifizierten Aktivitäten abgebildet und, wie in Abschnitt 3.1.2 beschrieben, mittels der Technik eines Affinitätsdiagrammes strukturiert. Das bedeutet, dass inhaltliche, aufeinander aufbauende Aktivitäten zusammenhängend dargestellt werden. Die oben beschriebenen Phasen (Oberthemen) werden durch die grauen Blöcke dargestellt (im Ausschnitt nicht sichtbar). Grüne Quadrate kennzeichnen die Aktivitäten des Usability Engineering und blaue Quadrate die des Software Engineering. Da es sich bei diesem Modell bereits um das finale

Modell dieser Arbeit handelt, existieren zudem noch magentafarbene und braune Quadrate. Diese werden im weiteren Verlauf der Arbeit näher erläutert.

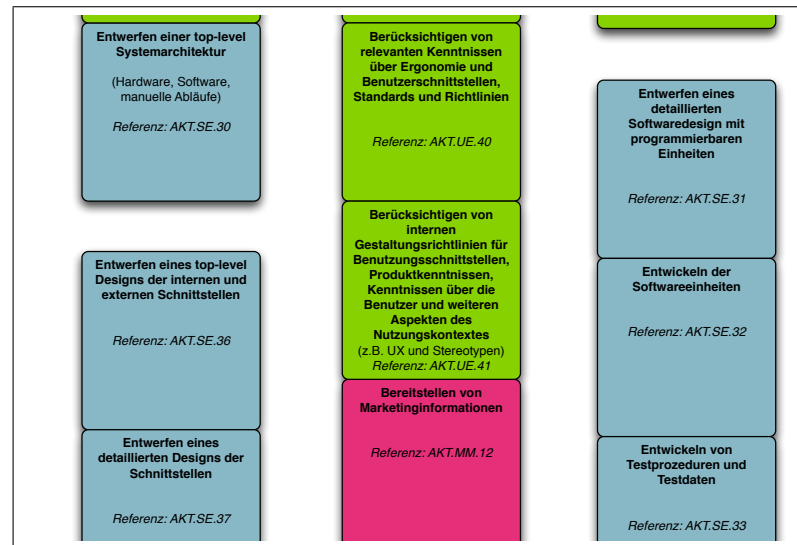


Abbildung 3.6.: Ausschnitt des Teilmodells der Aktivitäten

II) Teilmodell des UE Um die Abhängigkeiten zwischen den Aktivitäten und Artefakten des Usability Engineering zu verdeutlichen, wurde eine separate Sicht auf diese geschaffen (siehe Abbildung 3.7). Der zuvor angesprochen Lebenszyklus von Artefakten ist durch gerichtete Pfeile dargestellt. Zeigt ein Pfeil von einem Artefakt auf eine Aktivität, so dient dieses Artefakt als Input für die Aktivität. Zeigt der Pfeil von einer Aktivität auf ein Artefakt, so ist das Artefakt als Ergebnis dieser Aktivität entstanden.

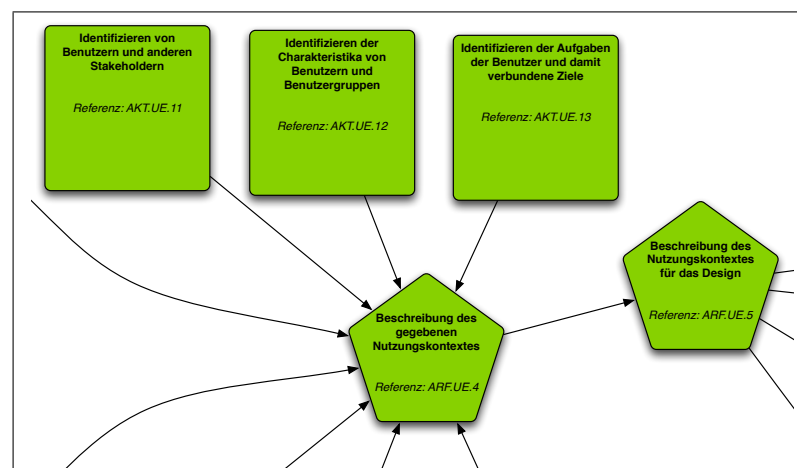


Abbildung 3.7.: Ausschnitt des Teilmodells des UE

III) Teilmodell des SE Analog zum Teilmodell des UE wurde des Weiteren auch eine Sicht auf Aktivitäten und Artefakte des Software Engineering modelliert (siehe Abbildung 3.8).

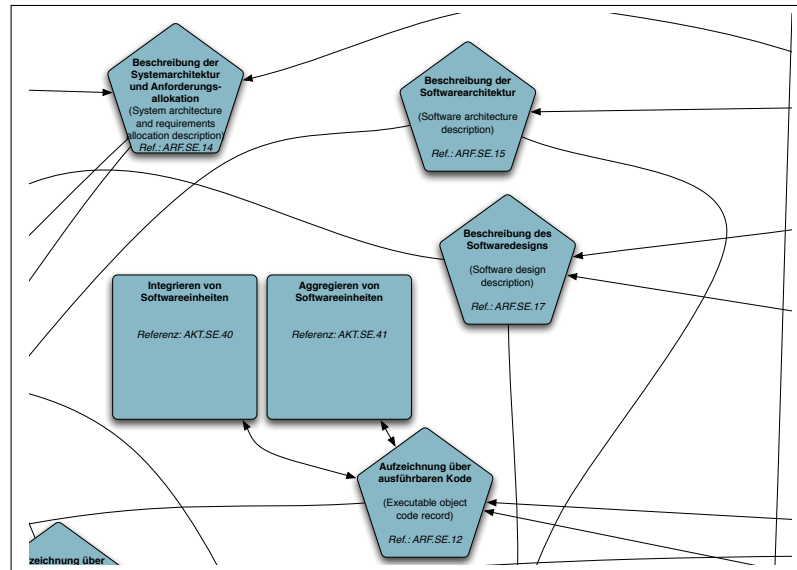


Abbildung 3.8.: Ausschnitt des Teilmodells des SE

IV) Teilmodell der Artefakte Ausgehend von den Abhängigkeiten zwischen Aktivitäten und Artefakten schien es interessant eine auf die Artefakte eingeschränkte Sicht zu erstellen (siehe Abbildung 3.9), um Wechselwirkungen zwischen den Phasen hervorzuheben. Unterschiedlich eingefärbte, gerichtete Pfeile zwischen den Artefakten kennzeichnen dabei, welche Artefakte für die Erstellung eines weiteren Artefaktes vorausgesetzt werden und verdeutlichen so die angesprochenen Wechselwirkungen.

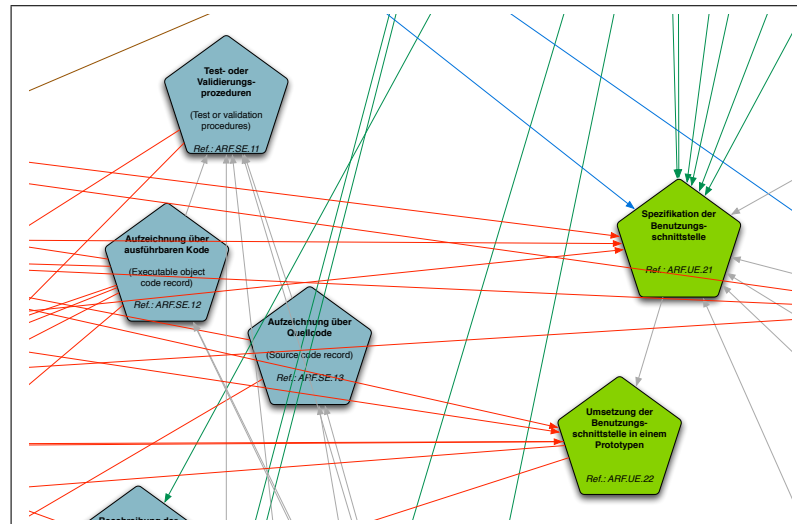


Abbildung 3.9.: Ausschnitt des Teilmodells der Artefakte

So wie ein Artefakt als Voraussetzung für ein Artefakt einer anderen Phase dient beziehungsweise durch dieses Einfluss auf ein Artefakt einer anderen Phase ausgeübt wird, wurde der gerichtet Pfeil farbig dargestellt. Dadurch sollten Wechselwirkungen zwischen den Phasen markiert werden. Die Bedeutung der Farbe wird in Tabelle 3.2 beschrieben.

Farbe	Phasen
Magenta ↔	Verstehen und Beschreiben des Nutzungskontextes Festlegen von Nutzungsanforderungen
Grün ↔	Festlegen von Nutzungsanforderungen Entwerfen von Gestaltungslösungen
Braun ↔	Festlegen von Nutzungsanforderungen Beurteilen von Gestaltungslösungen
Blau ↔	Verstehen und Beschreiben des Nutzungskontextes Entwerfen von Gestaltungslösungen
Rot ↔	Entwerfen von Gestaltungslösungen Beurteilen von Gestaltungslösungen
Petrol ↔	Verstehen und Beschreiben des Nutzungskontextes Beurteilen von Gestaltungslösungen

Tabelle 3.2.: Phasenübergreifende Abhängigkeiten der Artefakte

Eine wesentliche Erkenntnis, die aus den erstellten Teilmodellen gewonnen werden konnte, ist die des Lebenszyklus von Artefakten. Des Weiteren können durch die farbige Unterscheidung der gerichteten Pfeile die Wechselwirkungen zwischen den Phasen verdeutlicht werden (siehe Abbildung 3.10).

Sobald die Notwendigkeit einer benutzerorientierten Gestaltung in einem neuen Projekt festgelegt wurde, ist es aus Perspektive des Usability Engineering essentiell, Aktivitäten sämtlicher Phasen durchzuführen. So werden bereits in der Phase „*Verstehen und Beschreiben des Nutzungskontextes*“ Artefakte erzeugt, anhand derer erzeugte Gestaltungslösungen in Form von zum Beispiel Prototypen erst evaluiert werden können.

Soll zum Beispiel die Evaluationstechnik des Cognitive Walkthrough zur Beurteilung des entstandenen Prototypen angewandt werden, so bedienen sich die Experten der entwickelten Aufgabenmodelle des Nutzungskontextes und der Use Cases aus der Anforderungsspezifikation (vgl. Abbildung G.21 der erstellten Modelle im UE).

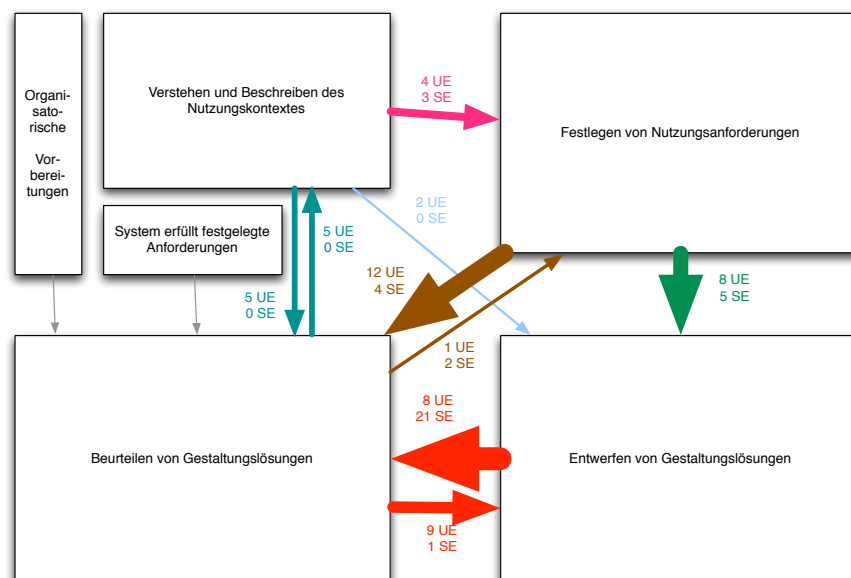


Abbildung 3.10.: Vereinfachtes Modell der Wechselwirkungen zwischen Phasen

Marketing & Marktforschung als weitere Disziplin? Bereits während der Analyse der Standards DIN EN ISO 13407 und ISO/IEC 12207 wurde die Erkenntnis zu Anknüpfungspunkten für eine weitere dritte Disziplin „Marketing und Marktforschung“ (siehe Kapitel 2.1.3) gewonnen. So lautet zum Beispiel eine Aussage in der Aktivität AKT.UE.41 „*Berücksichtigen von internen Gestaltungsrichtlinien für Benutzungsschnittstellen, Produktkenntnissen, Kenntnisse über die Benutzer und weiteren Aspekten des Nutzungskontextes*“, dass auch „Produktkenntnisse“ (ARF.UE.19) und „Marke-

tinginformationen“ (ARF.UE.20) mit einfließen.

Nach längerer Recherche und einem Gespräch mit einem Experten konnte festgehalten werden, dass für das Marketing und die Marktforschung keine etablierten Standards oder Vorgehensmodelle existieren, sondern dass es sich eher um kreative Prozesse handelt, bei denen es auf die Ergebnisse und nicht auf den Weg dorthin ankommt.

Anhand dieser Erkenntnis wurde ein Handbuch zu den *Grundlagen des Marketing* von Kotler et al. (2006) zur Identifizierung einiger passender Aktivitäten und Artefakte dieser Disziplin betrachtet.

Da die dabei gewonnen Informationen den im Usability Engineering definierten Nutzungskontext weiter detaillieren und auch zur Evaluation einer entstandenen Gestaltungslösung beitragen könnten, wurde nach Aktivitäten und Artefakten gesucht, um diese in das Modell der Integration einzugliedern.

Dabei konnten 16 Aktivitäten und zwei Artefakte (Bsp. siehe Abbildung 3.11) ermittelt und in das Modell eingebunden werden. Eine vollständige Auflistung dieser Aktivitäten und Artefakte ist in den Tabellen A.3 und A.6 in Anhang A zu finden. Die um die Aktivitäten und Artefakte erweiterten Modelle können in Abbildung G.1 und G.16 in Anhang G betrachtet werden.

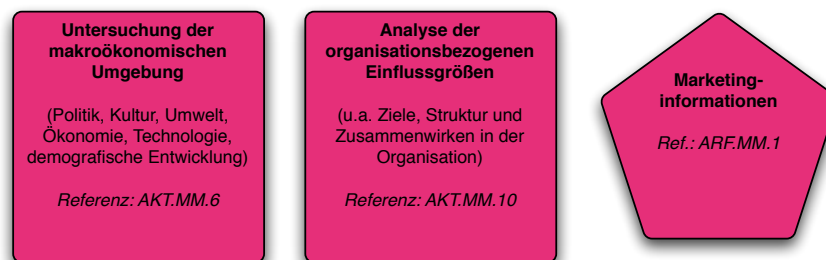


Abbildung 3.11.: Beispiele für Aktivitäten und Artefakte des Marketing und der Marktforschung

3.1.4. Intrinsische Validierung

Um die Plausibilität des entstandenen Modells zu überprüfen, ist es notwendig eine Validierung des Modells durchzuführen. So wurde die erste Stufe dieser Validierung intrinsisch anhand zweier Standards zur Prozessbewertung durchgeführt. Für das Usability Engineering wurde dabei die ISO/TS 18152 (2010) und für das Software Engineering die ISO/IEC 15504 (2006) verwendet. Beide Standards wurden bereits in Kapitel 2.3.2 und 2.3.3 vorgestellt. Zusammenfassend lassen sich die Standards anhand ihrer Beschreibungen wie folgt formulieren:

ISO/TS 18152 „*This Technical Specification presents a human-system (HS) model for use in ISO/IEC 15504-conformant assessment of the maturity of an organization in performing the processes that make a system usable, healthy and safe. It describes processes that adress human-system issues and the outcomes of these processes. It details the practices and work products associated with achieving the outcomes of each process.*“

ISO/IEC 15504 „*ISO/IEC 15504 provides overall information on the concepts of process assessment and its use in the two contexts of process improvement and process capability determination. It describes how the parts of the suite fit together, and provides guidance for their selection and use. It explains the requirements contained within ISO/IEC 15504, and their applicability to performing assessments.*“

Da bereits bei der Erstellung der Modelle die DIN EN ISO 13407 vollständig betrachtet wurde, wurde auch deren Standard zur Prozessbewertung ISO/TS 18152 nahezu vollständig, mit Schwerpunkt auf dem Sub-Prozess „*HS.3 Human-centred design*“, zur Validierung verwendet. Ebenfalls wie bei der ISO/IEC 12207 wurden auch bei deren Standard zur Prozessbewertung ISO/IEC 15504 hauptsächlich die Teile berücksichtigt, die sich auf den Entwicklungsprozess von Software beziehen. Anhand der relevanten Seitenzahlen der Standards in Bezug zu den gesamten Seitenzahlen, kann die Vollständigkeit der Betrachtung mit den in Tabelle 3.3 angegebenen Prozentsätzen beziffert werden.

ISO/TS 18152:	80%
ISO/IEC 15504:	62%

Tabelle 3.3.: Prozentuale Verwendung der Standards zur Prozessbewertung

Im Rahmen der Validierung wurden die im Modell verankerten Aktivitäten mit denen aus den Standards zur Prozessbewertung formulierten Base Practices (siehe Anhang B) abgeglichen. Dies geschah anhand der Beschreibungen der einzelnen Base Practices (BPs). So konnte der überwiegende Anteil der Aktivitäten durch ein oder mehrere entsprechende Base Practices belegt werden.

Zu sieben der insgesamt 28 Aktivitäten aus der DIN EN ISO 13407 konnten keine BPs bestimmt werden. Dementsprechend konnten somit 75% der Aktivitäten bestätigt werden. Bei den sieben unbestätigten Aktivitäten handelt es sich somit um Aktivitäten, die entweder nicht durch die ISO/TS 18152 abgedeckt werden oder die keine Relevanz in der Praxis besitzen. Um diese Frage zu klären bedarf es einer extrinsischen Validierung. Der Unterschied zu einer intrinsischen Validierung besteht darin, dass die Ergebnisse

nicht mit Standards abgeglichen werden, welche die Standards, aus denen die Ergebnisse ermittelt wurden, referenzieren. Stattdessen werden die Ergebnisse mit Experten „von außen“ (*extrinsisch*) validiert. Auf Basis der Durchführung des Abgleiches wurde ebenfalls betrachtet, in wie weit die Base Practices durch entsprechende Aktivitäten abgedeckt werden. Gegebenenfalls ist der Detaillierungsgrad der Base Practices höher als der in der DIN EN ISO 13407. So sollten Base Practices, die nicht auf die Aktivitäten der DIN EN ISO 13407 abgebildet werden konnten, dahingehend bewertet werden, in wie weit sich aus diesen neue Aktivitäten formulieren lassen, die in das Modell mit aufgenommen werden. Aus der Base Practice

„HS.1.2.BP7 - Assess the risks to the community and environment arising from human error in the use of the system“

wurde beispielsweise die für das Integrationsmodell ergänzende Aktivität

„AKT.UE.16 - Einschätzen von Auswirkungen auf die Gesellschaft und das Umfeld durch menschliche Fehler beim Gebrauch des Systems“.

In Tabelle B.1 im Anhang sind die Base Practices der ISO/TS 18152 gelistet und erweitert durch die Angabe, welche *Verwendung* der jeweiligen Base Practice im Modell zugeordnet wurde. Dabei wurde unterschieden, ob die Base Practice bereits durch eine Aktivität im Modell verankert ist (A), ob sich aus der Base Practice eine neue Aktivität formulieren lässt (N) oder ob es sich bei der Base Practice eher um eine organisatorische Rahmenbedingung handelte (O) (siehe Tabelle 3.4).

Referenz	Base Practice	Verv.	AKT. UE
HS.1.2.BP7	Assess the risks to the community and environment arising from human error in the use of the system	N	16
HS.1.4.BP2	Deliver training and other forms of awareness-raising to users and support staff.	O	-
HS.3.1.BP5	Describe the characteristics of any equipment external to the system and the working environment	A	14

Tabelle 3.4.: Ausschnitt Tabelle B.1 der Base Practices der ISO/TS 18152

Organisatorische Rahmenbedingungen (O) bezeichnen essentielle Aktivitäten, damit die Durchführung eines benutzerzentrierten Gestaltungsansatz gewährleistet werden kann. Da sich diese nicht direkt auf den inhaltlichen Prozess der Gestaltung auswirken, wurden diese nicht im Modell berücksichtigt. Anhand der durch in Tabelle 3.4 mit „N“ gekennzeichneten Base Practices, konnten weitere 11 neue Aktivitäten formuliert und in das Modell übernommen werden.

Äquivalent dazu konnten von den 65 identifizierten Aktivitäten aus der ISO/IEC 12207 66% mittels entsprechender Base Practices nachgewiesen werden. Dies entspricht 22 unbestätigten Aktivitäten, deren Beleg ebenfalls extrinsisch zu suchen ist. Zudem konnten aus den Base Practices weitere acht neue Aktivitäten benannt werden, so dass die Anzahl der Aktivitäten des Software Engineering auf 73 erhöht wurde (vgl. Tabelle 3.5).

Referenz	Base Practice	Verv.	AKT. SE
ENG.1.2.BP6	Update requirements	N	24
ENG.1.2.BP7	Communicate software requirements	N	25

Tabelle 3.5.: Ausschnitt Tabelle B.2 der Base Practices der ISO/IEC 15504

Eine vollständige Auflistung der Base Practices der ISO/IEC 15504 und deren Verwendung im Modell, kann in Tabelle B.2 im Anhang nachgelesen werden.

Sämtliche Aktivitäten und deren Abgleich zur den Standards der Prozessbewertung sind in Anhang A.1 und A.2 angegeben.

Anhand des Abgleiches der Aktivitäten durch entsprechende Base Practices konnten zudem die in Kapitel 3.1.3 adressierten und in Anhang D dokumentierten Gemeinsamkeiten zwischen Aktivitäten validiert werden. Dazu wurden die Aussagen in den Beschreibungen der Base Practices miteinander verglichen. Von den 16 identifizierten Gemeinsamkeiten konnten acht (50%) anhand der Aussagen in den BPs bestätigt werden. Drei Gemeinsamkeiten wurden auf Grund von keiner Übereinstimmung in den Aussagen wieder verworfen. Fünf Gemeinsamkeiten konnten wegen fehlenden BPs nicht belegt werden. Die Ergebnisse der Validierung der Gemeinsamkeiten wurden in Anhang D erfasst. Ein exemplarischer Auszug ist in Tabelle 3.6 dargestellt. So kann beispielsweise die Nachvollziehbarkeit von Anforderungen anhand der Aktivitäten und entsprechenden Base Practices als von beiden Disziplinen identisch durchgeführte Aktivität nachgewiesen werden.

Nr.	Disz.	Aktivität / BPs	Aussage	Auspräg.
G.10	UE	AKT.UE.26	<i>„The rationales, factors and weighting of human system issues for use in trade-offs should be documented so that they can be understood in the future.“</i>	identisch

Weiter auf der nächsten Seite

Nr.	Disz.	Aktivität / BPs	Aussage	Auspräg.
	SE	AKT.SE.17	„Traceability [...]“	
		ENG.1.BP2	„Define and implement the traceability process“	
		ENG.1.1.BP7	„Establish traceability“	
		ENG.1.3.BP5	„Establish traceability“	
		ENG.1.4.BP4	„Establish traceability“	

Tabelle 3.6.: Ausschnitt Tabelle D.1 der Gemeinsamkeiten zwischen Aktivitäten und deren Ausprägung

Auf Grund der Überarbeitung eines Standards wurden, im Anschluss an das ursprünglich erstellte Modell, Anpassungen durchgeführt. Diese werden im folgenden Abschnitt näher erläutert.

3.2. Anpassungen

Im zeitlichen Verlauf dieser Arbeit wurde der Standard DIN EN ISO 13407 durch den im März 2010 neu erschienenen und vollständig überarbeiteten Standard ISO 9241-210 ersetzt.

Um die Aktualität des erstellten Modells weiterhin zu gewährleisten, war eine Überarbeitung der Aktivitäten und Artefakte aus Perspektive des Usability Engineering unumgänglich.

Zunächst wurde der Standard auf seine Aussagen hin analysiert. Daraus wurden Tätigkeiten und Dokumentationsformen ermittelt und zu Aktivitäten beziehungsweise Artefakten überführt. Aus der Aussage

„[...] designing the sequence and timing (dynamics) of the interaction“

wurde dem entsprechend die Aktivität

„AKT.UE.36 - Gestalten von Sequenzen der Interaktion“.

Die so identifizierten Aktivitäten und Artefakte wurden allesamt tabellarisch festgehalten (siehe Anhang - Tabellen A.1 und A.4).

Im nächsten Schritt fand ein Abgleich der Aktivitäten und Artefakte der ISO 9241-210 zu denen der ehemaligen DIN EN ISO 13407 statt. So wurde überprüft, ob die jeweilige Aktivität beziehungsweise das jeweilige Artefakt bereits in der DIN EN ISO 13407 vorhanden war oder ob es sich um eine neue Aktivität beziehungsweise ein neues Artefakt handelt, welche/s mit in das Modell aufgenommen werden sollte. Die Referenz zur DIN EN ISO 13407 wurde dabei mit in einer Tabelle vermerkt. In Tabelle 3.7 ist ein

Ausschnitt des Ergebnisses zu sehen. Die vollständige Tabelle befindet sich in Anhang A.1 beziehungsweise A.2.

Referenz	UE-Phase	Aktivität	Quelle ISO 9241-210	Bezug ISO 13407
AKT.UE.11	VBN	Identifizieren von Benutzern und anderen Stakeholdern	9241-210:6.2.2.a	-
AKT.UE.12	VBN	Identifizieren der Charakteristiken von Benutzern und Benutzergruppen	9241-210:6.2.2.b	13407:7.2.1.a
AKT.UE.13	VBN	Identifizieren der Aufgaben der Benutzer und damit verbundene Ziele	9241-210:6.2.2.c	13407:7.2.1.b
AKT.UE.14	VBN	Identifizieren der Umgebung(en) des Systems (organisatorisch, sozial, kulturell, physisch)	9241-210:6.2.2.d	13407:7.2.1.c

Tabelle 3.7.: Referenzen der Aktivitäten des Usability Engineering (Ausschnitt)

Die Spalte *Referenz* nummeriert den jeweiligen Eintrag und besteht aus den zusammengesetzten Abkürzungen *AKT* für Aktivität und *UE* für Usability Engineering, sowie einer fortlaufenden Nummer. In der Spalte *UE-Phase* ist die Einordnung der Aktivität zur jeweiligen Phase des Usability Engineering vermerkt. Die Spalte *Aktivität* beinhaltet die Formulierung einer Aktivität. Die Referenzen zu den beiden Standards ISO 9241-210 und DIN EN ISO 13407 werden in den beiden letzten Spalten angegeben.

Anhand dieser Analyse konnten 25 neue Aktivitäten ermittelt werden, welche zuvor noch nicht auf Basis der DIN EN ISO 13407 erkannt wurden. Die bereits anhand der DIN EN ISO 13407 identifizierten Aktivitäten wurden vollständig bestätigt und sprachlich überarbeitet. Dem entsprechend konnte die Anzahl der Aktivitäten aus Perspektive des Usability Engineering im Modell nahezu verdoppelt werden. Die Experten konnten dem zu Folge die Aussagen innerhalb der ISO 9241-210 weiter präzisieren. Ebenfalls konnten auch 15 neue Artefakte ermittelt werden, welche im Modell verankert wurden.

Auf Grund der zahlreichen neuen Aktivitäten und Artefakte wurde im nächsten Schritt erneut eine intrinsische Validierung anhand der Base Practices der ISO/TS 18152 durchgeführt, um die Vollständigkeit des Modells nachzuweisen. Da durch die Analyse anhand der ISO 9241-210 die sprachlichen Formulierungen der Aktivitäten komplett überarbeitet wurden, konnten nicht nur vereinzelte Aktivitäten mit den Base Practices abgeglichen werden. Statt dessen wurden sämtliche Aktivitäten wiederholt bei der Durchführung des Abgleiches betrachtet.

Die Anzahl der Aktivitäten, die nicht durch eine entsprechende Base Practice belegt werden konnten, ist somit auf 15 angestiegen. Von den mittlerweile insgesamt 52 Akti-

vitäten konnten dem entsprechend 71,2% der Aktivitäten bestätigt werden. Der Bedarf einer extrinsischen Validierung zur Begründung der 15 nicht bestätigten Aktivitäten ist weiterhin vorhanden. Des Weiteren konnten sieben ergänzende Aktivitäten anhand der ISO/TS 18152 formuliert werden, zu denen keine Aussage in der ISO 9241-210 ermittelt werden konnte. Die Anzahl der Aktivitäten aus Perspektive des Usability Engineering ist somit auf 59 gestiegen. Im Vergleich zum Abgleich der DIN EN ISO 13407 mit der ISO/TS 18152 konnten mehrere Base Practices auf die Aktivitäten der ISO 9241-210 abgebildet werden. Von den 31 Artefakten im Modell konnten lediglich 17 Artefakte (54,8%) durch den Abgleich bewiesen werden. Eine vollständige Auflistung der Aktivitäten und Artefakte mit ihren Bezügen zur ISO/TS 18152 befindet sich im Anhang in Tabelle A.1 und A.4. Ebenso kann in Anhang B.1 die vollständige Auflistung der Base Practices und deren Verwendung im Modell betrachtet werden.

Des Weiteren wurden auch die zuvor identifizierten Gemeinsamkeiten zwischen Aktivitäten beziehungsweise Artefakten auf Grund der Änderungen erneut betrachtet (siehe Anhang - Tabellen D.1 und D.2). Als Fazit daraus lässt sich formulieren, dass keine identischen Aktivitäten und Artefakte, in Form einer redundant ausgeführten Tätigkeit, zwischen dem Usability Engineering und Software Engineering existieren. Dennoch bestehen einige Ähnlichkeiten zwischen Aktivitäten, die sich gegenseitig ergänzen. Es werden also „gleiche“ Aktivitäten jeweils aus einer unterschiedlichen Perspektive durchgeführt. So ist beispielsweise bei beiden Disziplinen der Bedarf der Nachvollziehbarkeit von Anforderungen und getroffenen Entscheidungen vorhanden. Im Standard ISO 9241-210 lautet eine Formulierung

„The rationales, factors and weighting of human system issues for use in trade-offs should be documented so that they can be understood in the future.“

während im Standard ISO/IEC 12207 die Aussage

„Traceability to the requirements and design of the software item.“

getroffen wird.

Auch zwischen den Artefakten existieren Ähnlichkeiten. So beschreiben beide Disziplinen ihre Anforderungen in einer Anforderungsspezifikation, wobei die Anforderungen aus Perspektive des Usability Engineering eher die Benutzer fokussieren, während der Fokus aus Perspektive des Software Engineering auf der Software und dem System liegt. Das heißt, die Sammlung von Anforderungen beider Disziplinen in einer gemeinsamen Anforderungsspezifikation könnte einen wesentlichen Anknüpfungspunkt zur Integration darstellen.

Bisher wurde das Modell lediglich intrinsisch validiert. Dabei ließ sich erkennen, dass nicht alle Aktivitäten und Artefakte validiert werden konnten beziehungsweise Aussagen getroffen wurden, die es zu bewerten gilt. Weitere Validierungsschritte sind also notwendig. Auf diese wird im nächsten Abschnitt eingegangen.

3.3. Validierung des Modells

Dieser Abschnitt beschäftigt sich mit der Konstruktvalidität des erstellten Integrationsmodells anhand von Konformitäts- und Rahmenanforderungen.

Die Konstruktvalidität bezieht sich auf Beständigkeit von Hypothesen, die aus dem Zielkonstrukt ableitbar sind und anhand von Testwerten bestätigt werden können (vgl. Bortz & Döring 2006). Das Zielkonstrukt ist auf Grund von Beobachtungen definiert und ermöglicht eine bestimmte Sicht auf die Realität (vgl. Kromrey 2009). Dieses Konstrukt ist in Form des Modells gegeben, welches eine Sicht auf die Integration von Usability Engineering und Software Engineering, sowie auf deren Aktivitäten und Artefakte erlaubt.

Zunächst wird der Bedarf der Validierung erläutert, bevor die Konformitäts- und Rahmenanforderungen, sowie der Abgleich des Modells mit diesen beschrieben wird.

3.3.1. Bedarf der Validierung

Validierung kann verstanden werden als eine „*Bestätigung durch Bereitstellung eines objektiven Nachweises*“ (ISO 9241-210). Dem entsprechend soll das Modell in Hinblick auf seine Vollständigkeit evaluiert werden. Bereits bei der intrinsischen Validierung, auf Basis von Standards zur Prozessbewertung, wurde deutlich, dass manche identifizierten Aktivitäten nicht durch eine entsprechende Base Practice belegt werden konnten. Um deren Existenz zu gewährleisten bedarf es einer extrinsischen Validierung. Diese soll zudem den Bezug des Modells zur Praxis sicherstellen, damit auch die Anwendbarkeit des Modells außerhalb der Theorie beurteilt werden kann.

Auf Grund der begrenzten Zeit zur Erstellung dieser Masterthesis wird die Validierung zunächst nur aus Perspektive des Usability Engineering durchgeführt.

3.3.2. Konformitäts- und Rahmenanforderungen

Als Grundlage zur Validierung wurden die von Nebe (2009) formulierten Konformitäts- und Rahmenanforderungen verwendet. Um Modelle des Software Engineering im Hinblick auf deren Fähigkeit zur Erstellung gebrauchstauglicher Produkte möglichst effektiv und effizient zu bewerten, führte Nebe (2009) eine expertenbasierte Erhebung

von Bewertungskriterien durch. Als Ziele dieser Erhebung betrachtete er die Überprüfung der Vollständigkeit und Korrektheit der Base Practices der ISO/TS 18152, die Bewertung von Prozessen, Aktivitäten, Ergebnissen und deren Qualität, sowie die Identifikation relevanter Aussagen zur Bewertung und Anwendung. Anhand von Interviews und Fragebögen erhielt er von den Experten 473 Aussagen. Diese wurden von ihm tabellarisch erfasst. Jede Aussage wurde daraufhin einer der vier Usability Engineering Aktivitäten („*Context of Use*“, „*User Requirements*“, „*Produce Design Solutions*“, „*Evaluation of Use*“) zugeordnet. Zudem wurden die Aussagen kategorisiert anhand ihrer inhaltlichen Ausrichtung, ob sie sich auf den Prozess als Ganzes, eine durchzuführende Aktivität oder ein Ergebnis beziehen. Zusätzlich wurden sie kategorisiert, ob die Aussagen auf ein Qualitätsmerkmal einer Aktivität abzielt oder ob eine Kritik am Prozess beziehungsweise Ergebnis durch die Aussage geäußert wurde.

Auf Basis der gesammelten und kategorisierten Aussagen konnte Nebe (2009) im nächsten Schritt 124 Konformitäts- und Rahmenanforderungen aus Sicht des Usability Engineering ableiten.

Ein Beispiel für eine solche Anforderung lautet:

„Die Kontextanalyse wird mit repräsentativen und geeigneten Nutzern (und mit Repräsentanten aller Nutzergruppen) durchgeführt.“

Diese wurden erzeugt, um sie an ein Vorgehensmodell des Software Engineering anzulegen und so zu bewerten, ob ein solches Modell die Perspektive des Usability Engineering berücksichtigt und somit zur Entwicklung gebrauchstauglicher Produkte verwendet werden kann. Insgesamt stellen diese Anforderungen eine Ergänzung zur ISO/TS 18152 dar. Dem entsprechend eignen sie sich auch zur Bewertung des in dieser Arbeit erstellten Integrationsmodells.

3.3.3. Abgleich

Im Rahmen der Validierung wurde das Modell mit den Konformitäts- und Rahmenanforderungen abgeglichen. Dazu wurde zunächst zu jeder Aktivität einzeln betrachtet, ob diese durch eine entsprechende Anforderung nach Nebe (2009) belegbar ist. Dabei konnten 43 von 59 Aktivitäten des Usability Engineering (entspricht 72,9%) bestätigt werden. Die 16 unbestätigten Aktivitäten haben ihren Ursprung hauptsächlich im Standard ISO 9241-210. Eine mögliche Erklärung dafür besteht demnach darin, dass dieser Standard zum Zeitpunkt der Entstehung der Konformitäts- und Rahmenanforderungen noch nicht existierte und somit auch noch keine Berücksichtigung im Standard ISO/TS 18152 gewährleistet sein kann.

Des Weiteren wurde auch aus Perspektive der Konformitäts- und Rahmenanforderungen überprüft, welche Anforderungen nicht durch eine Aktivität des Modells abgebildet

werden konnten. So konnten 35 von 124 Anforderungen keiner Aktivität zugeordnet werden. Nach einer genaueren Betrachtung ließen sich diese Anforderungen anhand ihrer Aussagen in drei Kategorien einordnen:

1. Prozessanforderung
2. Im Modell erfüllt
3. Neue Aktivität

Von den 35 nicht zugeordneten Anforderungen beziehen sich 22 auf die Durchführung beziehungsweise Sicherstellung einer Anforderung während des Prozesses (Kategorie 1).

Beispiel: „KoRa.A91 - *Zu allen relevanten Nutzeraspekten (z. B. Tätigkeiten, Rollen, Dokumentation, Personal) existiert eine Lösung.*“

Davon konnten zehn Anforderungen der zweiten Kategorie zugeordnet werden, da diese Anforderung im Modell erfüllt ist.

Beispiel: „KoRa.A51 - *Die Nutzeranforderungen sind die Basis für den nächsten Prozessschritt (PDS) und als solcher im Prozess verankert.*“

Die verbleibenden 13 Anforderungen wurden der dritten Kategorie zugewiesen. Sämtliche Konformitäts- und Rahmenanforderungen, sowie die Ergebnisse des Abgleiches sind im Anhang in Tabelle E.1 gelistet. Die daraus formulierten Anpassungen an das Modell werden im folgenden Abschnitt behandelt.

3.4. Angepasstes Modell

Aus den 13 zuvor nicht zugeordneten Konformitäts- und Rahmenanforderungen, die der dritten Kategorie „Neue Aktivität“ zugeordnet wurden, konnten 12 neue Aktivitäten für das Modell formuliert werden (siehe Tabelle 3.8).

Referenz	UE-Phase	Aktivität
KoRa.A9	VBN	Sicherstellen, dass die Kontextanalyse im richtigen, originären Kontext durchgeführt wird
KoRa.A11 KoRa.A39	VBN	Validieren der Kontextinformationen mit repräsentativen Nutzern und Kunden
KoRa.A12	VBN	Abbildern der Zusammenhänge zwischen Nutzern, Aufgaben, Zielen, Charakteristiken der Nutzeraufgaben und deren Umgebung, sowie den zugehörigen Rahmenbedingungen

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität
KoRa.A13	VBN	Sicherstellen, dass Kontextinformationen (für Dritte) verständlich und nachvollziehbar formuliert sind
KoRa.A18	VBN	Sicherstellen, dass Kontextinformationen Fakten abbilden und keine Interpretation der Situation darstellen
KoRa.A21	VBN	Abbilden der Kontextinformationen auf organisatorische bzw. Kundenziele
KoRa.A24	VBN	Sicherstellen, dass die Kontextinformationen die Sichtweise der Nutzer und keine technische Sichtweise repräsentiert
KoRa.A37	VBN	Demonstrieren der Ergebnisse für Projekt-Stakeholder
KoRa.A43	FNA	Sicherstellen, dass die Nutzungsanforderungen systemneutral (unabhängig einer Lösung) formuliert sind
KoRa.A61	FNA	Definieren von Kriterien zur Messung der Erreichung von Anforderungen
KoRa.A68	FNA	Definieren und vereinbaren von messbaren Kriterien zur Bewertung des Systems im Nutzungskontext
KoRa.A17	BGL	Berücksichtigen der Erfordernisse im Kontext als Validierungsgrundlage für die Lösung

Tabelle 3.8.: Neue Aktivitäten anhand der Konformitäts- und Rahmenanforderungen

Die erste Spalte beinhaltet die Referenz zur jeweiligen Konformitäts- und Rahmenanforderung (gelistet im Anhang - Tabelle E.1). Die Zuordnung zur Phase im Modell ist in der zweiten Spalte vermerkt. Die Abkürzungen setzen sich wie folgt zusammen:

- **VBN** - **V**erstehen und **B**eschreiben des **N**utzungskontext
- **FNA** - **F**estlegen von **N**utzungs**A**nforderungen
- **EGL** - **E**ntwerfen von **G**estaltungs**L**ösungen
- **BGL** - **B**eurteilen von **G**estaltungs**L**ösungen

Die jeweils aus der Anforderung formulierte Aktivität stellt den Inhalt der dritten Spalte dar. Die in Tabelle 3.8 definierten Aktivitäten konnten daraufhin in das Teilmodell der Aktivitäten (siehe Abbildung 3.12, braune Quadrate) eingebunden werden.

Da das Wissen, in Form ergänzender Aktivitäten und Artefakte, im Integrationsmodell erweitert und das Modell aus Perspektive des Usability Engineering zu 70 Prozent bestätigt werden konnte, kann diese Validierung als hilfreich betrachtet werden. Um die Konstruktvalidität des Modells nachzuweisen, bedarf es zusätzlichen Validierung, beispielsweise einer Expertenbefragung.

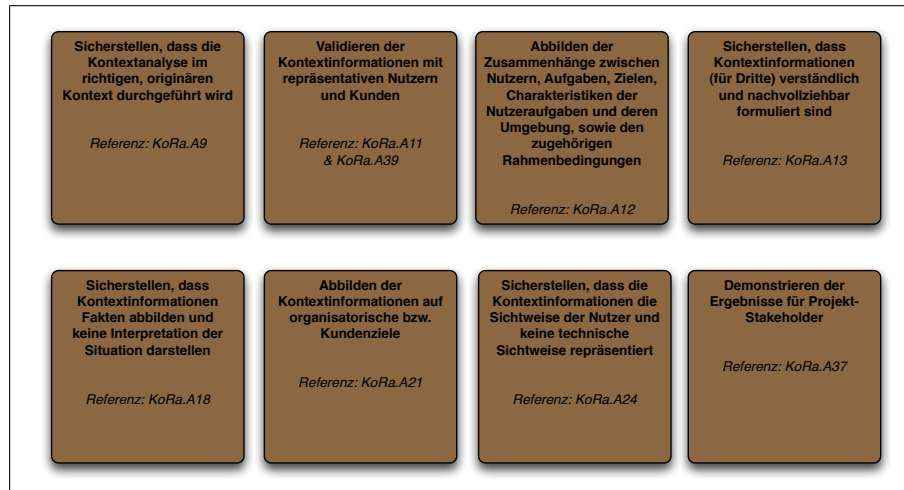


Abbildung 3.12.: Ausschnitt des Teilmodells der Aktivitäten

3.5. Zusammenfassung

In diesem Kapitel wurde zunächst die Erstellung eines ganzheitlichen Modells zur Integration von Usability Engineering und Software Engineering beschrieben. Anhand der Standards DIN EN ISO 13407 und ISO/IEC 12207 konnten aus beiden Perspektiven Aktivitäten und Artefakte ermittelt werden, die zur Entwicklung eines gebrauchstauglichen Produktes durchgeführt werden sollten. Daraufhin wurden die identifizierten Aktivitäten und Artefakte anhand der Phasen der DIN EN ISO 13407 in einem Modell angeordnet. Diese eignen sich auf Grund ihrer generischen Formulierung als Rahmenwerk des Modells, da sie nach Seffah et al. (2005b) auch den Prinzipien, „Analyse & Design“, „Entwicklung“ und „Evaluation“, des Software Engineering genügen.

Auf Grund von Wechselwirkungen zwischen Aktivitäten und Artefakten entsteht eine Komplexität, die dazu führte, dass bei der Digitalisierung das Modell in vier Teilmodelle aufgeteilt wurde. So entstand ein Teilmodell mit Sicht auf die Aktivitäten, jeweils ein Teilmodell mit Sicht auf die Aktivitäten und Artefakte des Usability Engineering und Software Engineering, sowie ein Teilmodell mit Sicht auf die Artefakte und deren Abhängigkeiten zueinander. Werden die Wechselwirkungen abstrahiert auf die Phasen des Modells, lässt sich erkennen, dass die Evaluation auf Artefakte der Kontextanalyse und der Anforderungsspezifikation zurückgreift, um die entstandene Lösung zu beurteilen. Des Weiteren suggeriert die Aufteilung der Phasen, dass eine Evaluation jeweils am Ende eine Iteration durchgeführt wird. Jedoch existieren Aktivitäten, in denen ausgesagt wird, dass die Ergebnisse bereits in jeder Phase selbst validiert werden sollten. Daher sollte eine Evaluationsphase, inklusive Validierungen, zentral dargestellt werden.

Eine weitere Disziplin, die sich in das ganzheitliche Modell integrieren ließe, ist die des Marketing und der Marktforschung, denn so wurden insbesondere aus Perspektive des Usability Engineering Aktivitäten identifiziert, die mit erzielten Informationen des Marketing und der Marktforschung verknüpft werden könnten.

Durch die Neuerscheinung der ISO 9241-210, welche die DIN EN ISO 13407 (1999) ersetzt, wurde das Modell aus Perspektive des Usability Engineering überarbeitet. So konnte die Anzahl der Aktivitäten und Artefakte des Usability Engineerings nahezu verdoppelt werden.

Eine daran anschließende intrinsische Validierung, in Form eines Abgleiches auf Basis der Standards ISO/TS 18152 (2010) und ISO/IEC 15504 (2006), konnte die Mehrheit der ermittelten Aktivitäten und Artefakte bestätigen. Jedoch blieben auch einige Aktivitäten unbestätigt, wodurch der Bedarf einer extrinsischen Validierung entstand. Die zuvor ermittelten Gemeinsamkeiten zwischen Aktivitäten und Artefakten wurden falsifiziert. Dies bedeutet, es existieren keine identischen Aktivitäten oder Artefakte, beispielsweise in Form von redundant durchgeführten Aktivitäten. Jedoch bestehen teilweise Ähnlichkeiten zwischen Aktivitäten beziehungsweise Artefakten, die sich gegenseitig ergänzen und so zu einer Steigerung der Qualität führen könnten.

Im Anschluss daran wurde das Modell aus Perspektive des Usability Engineering extrinsisch, anhand von Konformitäts- und Rahmenanforderungen nach Nebe (2009), validiert. Dazu wurden die Aktivitäten mit den Anforderungen abgeglichen. So konnten über 70 Prozent der Aktivitäten belegt und aus 13 Anforderungen neue Aktivitäten benannt werden.

Abschließend lassen sich, anhand des Modells und auf Grund der Durchführung zur Erstellung des Modells, acht Thesen formulieren.

1. Die vier Phasen der ISO 9241-210 eignen sich auf Grund ihrer Abstraktheit als Rahmenwerk für ein ganzheitliches Integrationsmodell.
2. Aktivitäten und Artefakte stehen in Wechselwirkung zueinander und bedingen sich gegenseitig (Artefaktlebenszyklus).
3. Die Analyse des Nutzungskontextes ist ausschlaggebend für das Verständnis des Problemraums und die Qualität der Lösung.
4. Eine gemeinsame Spezifikation von Anforderungen eignet sich als Ausgangsbasis für den weiteren Gestaltungs- und Entwicklungsprozess.¹⁴

¹⁴Voraussetzung wäre eine für alle beteiligten Personen verständliche Repräsentation der Anforderungen.

5. Eine Evaluation findet nicht nur am Ende jeder Iteration eines Gestaltungsprozesses statt, sondern sollte in jeder Phase durchgeführt werden.
6. Evaluationstechniken benötigen Modelle/Artefakte aller zuvor durchlaufenen Phasen als Bewertungsgrundlage.¹⁵
7. Usability Engineering und Software Engineering besitzen keine identischen Aktivitäten oder Artefakte.¹⁶
8. Artefakte beider Disziplinen gewinnen durch eine Integration gegenseitig an Qualität.¹⁷

Die Konstruktvalidität des Integrationsmodells konnte bisher nur ansatzweise nachgewiesen werden. Deshalb ist eine weitere Validierung, in Form einer Befragung durch Experten, notwendig. Diese ist Thema des nächsten Kapitels.

¹⁵Bspw. werden für die Technik des „Cognitive Walkthrough“ die Aufgabenmodelle aus der Beschreibung des Nutzungskontextes benötigt.

¹⁶„Identisch“ bedeutet redundant durchgeführte Aktivitäten bzw. Artefakte. Eine Anforderungsspezifikation wird in dieser Arbeit als „ähnlich“ betrachtet, da diese je nach Disziplin andere „Typen“ von Anforderungen enthalten und ggf. auch unterschiedlich repräsentiert werden.

¹⁷Bspw. durch die Berücksichtigung der Spezifikation der Benutzerinteraktion bei den Überlegungen zur Kommunikation in der Systemarchitektur

4. Expertenbefragung

Nachdem das Integrationsmodell anhand von Konformitäts- und Rahmenanforderungen validiert und anschließend angepasst wurde, wird eine Expertenbefragung durchgeführt. Durch die Aussagen der Befragung als induktives Verfahren, soll vom Konkreten (dem Wissen der Experten) auf das Abstrakte (das Integrationsmodell) geschlossen werden (vgl. Bortz & Döring 2006). Dadurch sollen weitere Erkenntnisse über die Validität des Modells aufgezeigt und neues Wissen im Modell ergänzt werden. Alternativ dazu existieren deduktive Verfahren. Dabei wird vom Abstrakten auf das Konkrete geschlossen. So werden beispielsweise aus Theorien Hypothesen deduziert, die es dann zu falsifizieren gilt. Verifikationen sind dabei nicht möglich. Ergänzend zur wahrheitsbewahrenden und weniger innovativen Deduktion und zur wahrheitserweiternden, aber unsicheren Induktion, existiert die Abduktion. Diese erweitert den Geltungsbereich von Fakten, indem versucht wird diese zu interpretieren. Abduktionsschlüsse werden daher als stark spekulativ angesehen. Auf Grund der Entscheidung neues Wissen zu generieren, um das Modell zu erweitern, wurde ein induktives Verfahren ausgewählt. Dazu wird zunächst in diesem Kapitel die Methodik zur Formulierung der Fragen, inklusive der Operationalisierung des Untersuchungsgegenstandes, sowie der dimensional und semantischen Analyse erläutert (vgl. Kromrey 2009). Des Weiteren wird die Durchführung der Befragung betrachtet. Dabei wird erklärt, wie die einzelnen Fragestellungen entstanden sind, wie die Befragung geplant und wie die Experten ausgewählt wurden. Die Befragung wurde mit zwei Experten des Usability Engineering realisiert. Eine umfangreichere Befragung, inklusive von Experten des Software Engineering, war auf Grund des begrenzten zeitlichen Rahmens dieser Arbeit nicht möglich. Abschließend werden die Ergebnisse ausgewertet und die daraus gewonnenen Erkenntnisse auf das Modell angewandt. Dazu wurden sowohl die quantitativen Bewertungen der Aktivitäten und Artefakte, als auch die qualitativen Aussagen der Experten betrachtet. Eine quantitative Auswertung ist auf Grund der geringen Anzahl von Interviewpartnern nur bedingt ausdrucksstark, sofern beide Experten keine gegenteilige Bewertung abgeben.

4.1. Methodik

Im vorherigen Kapitel konnte die Konstruktvalidität des Modells zunächst ansatzweise nachgewiesen werden. Als weiterer Schritt soll zur Validierung eine Expertenbefragung durchgeführt werden.

Ziele dieser expertenbasierten Befragung sind folgende:

1. Nachweis der Akzeptanz und Praktikabilität von Aktivitäten des Usability Engineering, so dass der Bezug des Modells zur Praxis und somit die Sinnhaftigkeit dessen begründet ist.
2. Überprüfung der Vollständigkeit und Korrektheit der identifizierten Aktivitäten, Artefakte und Wechselwirkungen.
3. Nachweis, der anhand des Integrationsmodells formulierten Thesen (siehe Kapitel 3.4).

Geplant ist eine qualitative teilstandardisierte Befragung in Form eines Leitfragengesprächs. Dies besteht nach Kromrey (2009) hauptsächlich aus offenen Fragen, wodurch der Interviewer die Möglichkeit bekommt, die Befragung selbst mit zu strukturieren, um zu bestimmten Themen genauer nachzufragen. Zudem fördert die Teilstandardisierung ehrliche Antworten des Interviewten, da keine Antwortmöglichkeiten vorgegeben sind, wodurch der Interviewte nicht auf die Meinung des Interviewers schließen kann (vgl. Bortz & Döring 2006).

Offene Fragen als Leitfragen geben dem Interviewer mehr oder weniger verbindlich die Art und die Inhalte des Gesprächs vor. Durch mehrere Befragungen lassen sich zudem anhand der Leitfragen die Aussagen der Interviewten in einer gewissen Struktur miteinander vergleichen.

Der Anspruch an die Autorität ist neutral. Dadurch wird die informationssuchende Funktion des Interviewers betont und der Interviewte wird als gleichwertiger Partner gesehen. Die Funktion des Interviews ist informatorisch, damit die Tatsachen deskriptiv erfasst werden können. Daher wird das Interview in Form eines Einzelinterviews durchgeführt, damit sich die Aussagen, auf Grund von anderen Interviewten, nicht verfälschen.

Die Entwicklung der Leitfragen sieht vor, den Aufbau auf Makro- und Mikroebene zu planen (vgl. Bortz & Döring 2006). Während die Makroplanung die Abfolge der einzelnen thematischen Teilbereiche festlegt und somit die Struktur der Befragung definiert,

spezifiziert die Mikroplanung die genauen Inhalte, die zu den einzelnen Themenbereichen erfragt werden sollen und präzisiert die Formulierung der Fragen.

Um die zu erfragenden Themen messbar zu gestalten, ist zunächst eine Operationalisierung des Themengebietes auf Basis einer dimensional Analyse notwendig. Im Rahmen einer dimensionale Analyse werden nach Kromrey (2009) die Bereiche („Dimensionen“) der Realität identifiziert, die durch die Forschungsfragestellung explizit angesprochen werden. Dimensionen bezeichnen diejenigen Merkmale, nach denen empirische Sachverhalte unterschieden werden können. Des Weiteren wird beurteilt, ob die als relevant angenommenen Dimensionen zusammengefasst werden, oder ob diese differenziert betrachtet werden. Im Anschluss werden geeignete, objektsprachliche Begriffe gewählt, die den Untersuchungsgegenstand in einer angemessenen Differenzierung abbilden. Anhand der Dimensionen wird daraufhin der Fragegegenstand formuliert. Um eine präzise Formulierung des Fragegegenstandes sicherzustellen und so Mehrdeutigkeiten in der Formulierung zu vermeiden, bedarf es einer zusätzlichen semantischen Analyse. Dabei wird die Bedeutung von Begriffen, auf Basis einer Rekonstruktion ihrer „semantischen Regeln“ und „empirischen Interpretation“, erschlossen. Die dimensionale bzw. semantische Analyse, sowie die Präzisierung der Fragestellung sind für den erfolgreichen Forschungsprojektverlauf kritische Punkte. Diese können in der späteren Untersuchungsphase nicht mehr zurückgenommen bzw. Unterlassungen nicht mehr korrigiert werden. (vgl. Kromrey 2009)

Die Erarbeitung der Dimensionen und Fragen, sowie die Planung der Befragung ist Gegenstand im nächsten Abschnitt.

4.2. Durchführung

Anhand der zuvor erläuterten Methodik, werden im ersten Schritt die Dimensionen des Untersuchungsgegenstandes formuliert. Den Untersuchungsgegenstand stellt das in dieser Arbeit entwickelte Modell zur Integration von Usability Engineering und Software Engineering dar.

Auf Basis der Ziele dieser Befragung lassen sich zunächst vier Aspekte identifizieren, anhand derer das Modell durch Experten beurteilt werden soll.

- Akzeptanz
- Vollständigkeit
- Abhängigkeiten
- Thesen

Akzeptanz: Die Akzeptanz bezieht sich darauf, inwieweit das Modell beziehungsweise dessen Thematik der Integration von Usability Engineering und Software Engineering in der Praxis akzeptiert werden könnte. Interessant dabei ist, welche Relevanz der Aspekt „Usability“ in den Unternehmen spielt. Insbesondere in Bezug auf deren strategische Ausrichtung, deren operatives Geschäft und deren Unternehmenserfolg. Des Weiteren wäre eine Einschätzung aufschlussreich, inwieweit benutzerorientierte Gestaltungsansätze in der Praxis Verwendung finden. Dementsprechend werden die zwei Dimensionen „Usability“, als Produkteigenschaft, und „Human-Centred Design“, als Prozess, betrachtet.

Vollständigkeit: Da das Modell aus „Phasen“, „Aktivitäten“ und „Artefakten“ besteht, bezieht sich die Vollständigkeit auf diese Dimensionen. Dabei soll durch die Experten beurteilt werden, ob eine vorgegebene Liste von Aktivitäten beziehungsweise Artefakten ergänzt werden sollte. Gegebenenfalls können auch redundante Aktivitäten und Artefakte ermittelt und von der Liste entfernt werden. Des Weiteren soll die Bedeutung der einzelnen Aktivitäten und Artefakte bewertet werden.

Abhängigkeiten: Der Aspekt der Abhängigkeiten oder Wechselwirkungen soll anhand der Dimension der „Phasen“ betrachtet werden. Die Experten sollen dabei eine Einschätzung über die Abhängigkeiten zwischen Phasen geben und notwendige Informationen für den Beginn einer Phase festlegen.

Thesen: Bereits in Kapitel 3.5 wurden acht Thesen formuliert. Deren Wahrheitsgehalt soll durch die Experten geprüft werden. Anhand der Thesen ergeben sich die Dimensionen „Generische Phasen“, „Artefaktlebenszyklus“, „Nutzungskontext“, „Artefakte“, „Evaluation“, „Evaluationstechniken“ und „Aktivitäten“.

Auf Basis der dimensionalen Analyse, vorgestellt in Kapitel 4.1, wird zu jeder Dimension mindestens ein Fragegegenstand präzise beschrieben, um daraufhin die eigentlichen Fragen zu formulieren. Unter dem Aspekt der „Akzeptanz“ und der Dimension des „Human-Centred Design“ wird so zum Beispiel der Fragegegenstand

„Einschätzung, inwieweit benutzerorientierte Gestaltungsansätze in der Praxis Verwendung finden“

festgelegt. Daraus wird dann die konkrete Frage

„Wie hoch schätzen Sie die Verbreitung von Entwicklungsansätzen in der Praxis ein, deren Fokus auf den Benutzern liegt?“

abgeleitet. Eine vollständige Betrachtung der Ergebnisse der dimensionalen Analyse kann im Anhang in Tabelle F.1 eingesehen werden.

Da das mentale Modell, und somit auch die Antworten der noch verbleibenden Fragen, mit jeder gestellten Frage beeinflusst wird, ist es notwendig, die Reihenfolge der Fragen dementsprechend zu strukturieren. Dabei dürfen dem Interviewten die Fragen zur Bewertung der identifizierten Aktivitäten beispielsweise erst dann gestellt werden, nachdem dieser zunächst die Aktivitäten einer Phase selbst definiert hat.

Der nach der Strukturierung aufgestellte Befragungsplan ist in Tabelle 4.1 dargestellt. Auf Makroebene werden die Themen der einzelnen Inhaltsblöcke anhand der Aspekte der dimensional Analysis festgelegt. Auf Mikroebene werden die Fragen sortiert und mit einer fortlaufenden Nummer versehen.

Thema	Nr	Frage
Einstieg / Akzeptanz von Usability	F.1	Wie schätzen Sie die Relevanz von Usability in den Unternehmen ein, bezogen auf deren strategische Ausrichtung? (Likert Skala: sehr hoch, hoch, neutral, niedrig, sehr niedrig)
	F.2	Wie schätzen Sie die Relevanz von Usability in den Unternehmen ein, bezogen auf das operative Geschäft? (Likert Skala: sehr hoch, hoch, neutral, niedrig, sehr niedrig)
	F.3	Wie schätzen Sie die Relevanz von Usability in den Unternehmen ein, bezogen auf deren Erfolg? (Likert Skala: sehr hoch, hoch, neutral, niedrig, sehr niedrig)
	F.4	Wie hoch schätzen Sie die Verbreitung von Entwicklungsansätzen in der Praxis ein, deren Fokus auf den Benutzern liegt? (Likert Skala: sehr hoch, hoch, neutral, niedrig, sehr niedrig)
Vollständigkeit	F.5	Wie beurteilen Sie die Vollständigkeit der in der ISO 9241-210 definierten Phasen zur Durchführung einer benutzerorientierten Entwicklung?
	F.6	Wie beurteilen Sie die zeitliche Anordnung der Phasen in der ISO 9241-210?
	F.7	Bewerten Sie die in der folgenden Liste aufgeführten Phasen der ISO 9241-210. (Likert Skala: sehr bedeutend, bedeutend, neutral, weniger bedeutend, unbedeutend)
	F.8	Welche Wechselwirkungen existieren zwischen den vier Phasen der ISO 9241-210?
F.9 - F.17 je Phase	F.9	Anhand welcher Kriterien lässt sich der Erfolg der Phase [...] messen?
	F.10	Welche Indikatoren kennzeichnen das Ende einer Phase [...]?
	F.11	Aus welchen Aktivitäten setzt sich die Phase [...] zusammen?
	F.12	Betrachten Sie folgende Liste von Aktivitäten und ergänzen Sie diese falls notwendig.
	F.13	Bewerten Sie die in der folgenden Liste aufgeführten Aktivitäten.

Weiter auf der nächsten Seite

Thema	Nr	Frage
		(Likert Skala: sehr bedeutend, bedeutend, neutral, weniger bedeutend, unbedeutend)
	F.14	Welche Informationen sind notwendig für den Beginn der Phase [...]?
	F.15	In welcher Form werden die Ergebnisse der Phase [...] dokumentiert?
	F.16	Betrachten Sie folgende Liste von Dokumentationsformen und ergänzen Sie diese falls notwendig.
	F.17	Bewerten Sie die in der folgenden Liste aufgeführten Dokumentationsformen. (Likert Skala: sehr bedeutend, bedeutend, neutral, weniger bedeutend, unbedeutend)
Thesen	F.18	Würden Sie behaupten, dass eine Evaluation nicht nur zum Abschluss einer Iteration, sondern bereits nach jeder Phase durchgeführt werden sollte?
	F.19	Würden Sie behaupten, dass die Beschreibung des Nutzungskontextes und die Anforderungsspezifikation essentiell für die Durchführung gewisser Evaluationstechniken (wie z.B. „Cognitive Walkthrough“, „Think Aloud“ oder „Usability Testing“) sind?
	F.20	Wie beurteilen Sie das Ergebnis der Evaluation eines existierenden Produktes, wenn dazu keine Beschreibung des Nutzungskontextes und keine Anforderungsspezifikation vorliegt?
	F.21	Wie schätzen Sie die Bedeutung einer Analyse des Nutzungskontextes in Bezug auf das Verständnis des Problemraumes ein?
	F.22	Wie beurteilen Sie den Zusammenhang zwischen dem genauen Verständnis des Problemraumes und der Qualität der Lösung?
	F.23	Würden Sie behaupten, dass die vier generischen Phasen der ISO 9241-210 auch in der Disziplin des Software Engineering für den übergeordneten Lebenszyklus eines Systems akzeptiert werden?
	F.24	Würden Sie behaupten, dass ohne die Integration des UE und SE unabhängig voneinander Aktivitäten durchführen die sich gleichen?
	F.25	Würden Sie behaupten, dass durch einen Informationsaustausch an im Prozess fest definierten Punkten die Qualität der jeweils von UE und SE erstellten Ergebnisse gesteigert wird?
	F.26	Wie sehen Ihrer Meinung nach Anknüpfungspunkte zwischen den Disziplinen aus?
	F.27	Würden Sie behaupten, dass UE und SE durch eine gemeinsam angefertigte Anforderungsspezifikation die Qualität der daraus resultierenden Ergebnisse steigern können?

Tabelle 4.1.: Befragungsplan

Die meisten Fragen sind offen gestellt und geben keine Antwortmöglichkeit vor, um den Interviewten nicht in seiner Aussage zu beeinflussen und um die angesprochene

Ehrlichkeit in der Antwort zu fördern.

Fragen F.1 bis F.4 erhalten zusätzlich einen Bewertungsbogen (siehe Abbildung 4.1), auf dem die Experten die Relevanz anhand der von Likert (1932) entwickelten fünfstufigen Skala einschätzen sollen. Ziel ist es, eine eindeutige, messbare Aussage zu erhalten. Diese Skala deckt dabei sowohl positive Extrema (bspw. *sehr hoch*), als auch negative Extrema (bspw. *sehr niedrig*) ab und ermöglicht zudem eine neutrale Bewertung. (vgl. Bortz & Döring 2006)

1) Wie schätzen Sie die Relevanz von Usability in den Unternehmen ein, bezogen auf deren strategische Ausrichtung?

☐ sehr hoch
 ☐ hoch
 ☐ neutral
 ☐ niedrig
 ☐ sehr niedrig

Abbildung 4.1.: Bewertungsvorlage Fragen F.1 - F.4

Fragen F.9 bis F.17 werden wiederholt für jede Phase gestellt, um so die Aufmerksamkeit auf die jeweilige Phase zu erhalten. Bei den Fragen F.12 und F.16 bekommen die Experten die für die Phase zutreffende Liste mit Aktivitäten beziehungsweise Artefakten vorgelegt. So sollen diese ergänzt, gekürzt und unter Umständen auch sprachlich kommentiert werden. Jeweils daran anschließend wird die Liste mit denselben Aktivitäten oder Artefakten vorgelegt, erweitert um eine Likert Skala (siehe Abbildung 4.2). Die Gewichtung soll in der Auswertung eine zusätzliche Information über die Relevanz der einzelnen Aktivitäten und Artefakte liefern.

Bei der eigentlichen Durchführung der Befragung wurden zwei Experten des Usability Engineering jeweils über eineinhalb Stunden in einem Einzelinterview befragt. Diese Experten sind bereits seit über zehn Jahren als Usability Engineer tätig und wirken aktiv in Arbeitskreisen zu Qualitätsstandards in der Softwareentwicklung mit. Über persönliche Kontakte konnten diese für die Beurteilung dieser Arbeit gewonnen werden. Auf Grund der begrenzten Zeit konnten lediglich die Fragen F.1 bis F.17 im persönlichen Gespräch beantwortet werden. Die Fragen F.18 bis F.27 wurden den Experten in schriftlicher Form zur Beantwortung zugesandt.

Die Auswertung der Ergebnisse dieser Expertenbefragung wird im nächsten Abschnitt behandelt.

Bewertungsvorlage: AKTIVITÄTEN // Experte E_____

Bewerten Sie die in der folgenden Liste aufgeführten Aktivitäten.

Nr	Aktivität - Phase 1	sehr bedeutend	bedeutend	neutral	weniger bedeutend	unbedeutend
AKT. UE.10	Definieren der Abgrenzung (Scope) des Nutzungskontextes	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>
AKT. UE.11	Identifizieren von Benutzern und anderen Stakeholdern	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>	<input type="text"/>

Abbildung 4.2.: Beispiel einer Bewertungsvorlage für Fragen F.7, F.13 & F.17

4.3. Resultate

Auf der Basis der durch die Experten in den Befragungen erhaltenen Aussagen (siehe Anhang F - Tabelle F.5), wird in diesem Abschnitt eine Einschätzung zur Validität des Modells gegeben.

Zunächst wird die Relevanz des Aspektes „Usability“ eingeschätzt (Fragen F.1 bis F.4). Neben einer qualitativen Aussage zu den Fragen, wurde zusätzlich eine quantitative Bewertung durch die Experten abgegeben (siehe Tabelle 4.2). Die quantitative Bewertung ist auf Grund der Anzahl der Interviewpartner nur bedingt aussagekräftig, sofern beide Experten in ihrer Bewertung stark voneinander abweichen. Auf der strategischen Ebene eines Unternehmens beurteilen die Experten die Relevanz als „hoch“. Auch wenn Usability nicht nach außen hin kommuniziert wird, so ist es trotzdem Vorstandsthema. Jedoch ist dies stark von der jeweiligen Domäne der Unternehmen abhängig. Während in den softwareherstellenden Unternehmen der Aspekt Usability, besonders seit der Aufnahme des Begriffes „User Experience“ in das Vokabular, faktisch ein „*großer strategischer Pfosten*“ ist, so muss der Aspekt in den softwareanwendenden Unternehmen noch weiter etabliert werden. Dort werden besonders die Themen „Effizienzsteigerung“, „Konsensbildung“, „Kundengewinnung“, sowie „Kundenbindung“ betrachtet.

Nr	Frage	sehr hoch	hoch	neutral	niedrig	sehr niedrig
F.1	Wie schätzen Sie die Relevanz von Usability in den Unternehmen ein, bezogen auf deren strategische Ausrichtung?		2			
F.2	Wie schätzen Sie die Relevanz von Usability in den Unternehmen ein, bezogen auf das operative Geschäft?		1	1		
F.3	Wie schätzen Sie die Relevanz von Usability in den Unternehmen ein, bezogen auf deren Erfolg?		1		1	
F.4	Wie hoch schätzen Sie die Verbreitung von Entwicklungsansätzen in der Praxis ein, deren Fokus auf den Benutzern liegt?				2	

Tabelle 4.2.: Einschätzung der Experten zur Relevanz von Usability und Human-Centred Design (HCD) im Unternehmen

Bezogen auf das operative Geschäft wird Usability von den Experten „neutral bis hoch“ bewertet. Usability ist nach wie vor ein Aspekt von vielen und gilt nicht als selbstverständlich. Je mehr IT-Unterstützung im Unternehmen notwendig ist (zum Beispiel um den asynchronen Austausch von Informationen zwischen Innen- und Außendienst zu gewährleisten), desto höher ist die Relevanz von Usability.

Der Unternehmenserfolg anhand von Usability wird durch die Experten „niedrig bis hoch“ eingeschätzt. Ausschlaggebend dabei ist die Branche in der das jeweilige Unternehmen tätig ist. Haben die entwickelten Produkte einen starken Bezug zum Kunden, ist die Relevanz von Usability, bezogen auf den Wettbewerbsvorteil, sehr hoch. Werden vorwiegend Produkte für den internen Gebrauch im Unternehmen entwickelt, so wird Usability eher als unwichtig angesehen.

Die Verbreitung von benutzerorientierten Entwicklungsansätzen in der Praxis wird von den Experten als „niedrig“ beurteilt. Die Unternehmen, die sich mit dem Aspekt Usability auseinandersetzen, versuchen zwar an den Nutzer zu denken, doch die Wenigsten davon führen einen systematischen Ansatz durch. Teilweise liegt dies auch an einem Mangel von qualifiziertem Personal.

Usability scheint in den Unternehmen sowohl auf der strategischen, als auch auf der operativen Ebene Aufmerksamkeit zu erlangen. In Bezug auf diese Arbeit bedeutet dies, dass eine Integration von Usability Engineering und Software Engineering durchaus zu einer Verbesserung bei der systematischen Durchführung in der Entwicklung gebrauchstauglicher Produkte beitragen kann.

Die Vollständigkeit des Modells, im Hinblick auf die Phasen der ISO 9241-210, wird durch die Experten bestätigt. In Bezug auf Usability sind diese ausreichend benannt. Gegebenenfalls könnten die Phasen um die „Organisation“ erweitert werden. Dazu zählen Tätigkeiten wie die Entwicklungsbegleitung und die Überprüfungen, ob unternehmensinterne Styleguides eingehalten werden, sowie Aspekte, wie Vergleichbarkeit, Konsistenz oder Integrierbarkeit. Ob diese allerdings im Usability Engineering verankert sind oder von einer darüber liegenden organisatorischen Ebene im Unternehmen behandelt werden, ließen die Experten offen.

Jedoch sind sie der Auffassung, dass es sich bei dem in der ISO 9241-210 dargestellten Modell nicht um ein Phasenmodell handelt, da Aktivitäten nicht unbedingt sequenziell, sondern vielmehr parallel durchgeführt werden. Vielmehr könnte der tatsächliche Ablauf als Wellenmodell beziehungsweise in Form von versetzten Sinuskurven beschrieben werden, da sämtliche Phasen über den gesamten Projektverlauf ausgeführt werden. Das Modell im Standard stellt lediglich die Abhängigkeiten dar. Der Einstieg in dieses Modell ist jeweils abhängig von der Art des Projektes. Bei neuen Projekten („*grüne Wiese*“) bietet der durch das Modell suggerierte Start mit der Analyse des Nutzungskontextes einen angemessenen Einstieg. Soweit jedoch ein existierendes System überarbeitet werden soll, bietet es sich an, zunächst das bestehende System zu beurteilen, um so bereits neue Anforderungen aus den Erfahrungen mit dem alten System abzuleiten. Handelt es sich um ein Innovationsprojekt, so ist häufig die Lösungsidee der Innovationstreiber und nicht das Verständnis des Problems.

Insgesamt betrachtet, kann der durch das Modell in der ISO 9241-210 beschriebene logische Fluss bestätigt werden. In einem Projekt wird dabei immer mit den Ergebnissen einer Vorphase gearbeitet, auch wenn diese nicht durchgeführt wurde. In diesem Fall werden Antizipationen getroffen. Die Experten formulieren, dass die „Wahrheit“ letztendlich immer durch die Evaluation aufgezeigt wird. Aus den Erfahrungen zeigt sich, je weniger Budget in der Analyse ausgegeben wird, desto mehr muss im Abschluss eines Projektes evaluiert und überarbeitet werden. Alle Phasen wurden somit als „bedeutend bis sehr bedeutend“ gekennzeichnet und auch die Wechselwirkungen zwischen den Phasen konnten validiert werden.

Mittels der Fragen 9 bis 17 wurden die Vollständigkeit und die Abhängigkeiten in Bezug auf die Aktivitäten und Artefakte des Modells beurteilt. Abhängigkeiten in der Hinsicht, dass gewisse Informationen einer Phase für den Beginn der nächsten Phase zur Verfügung stehen, konnten im Modell durch die Experten bestätigt werden. Auf Grund der Aussage, dass „ausreichend valide Daten des Nutzungskontextes zur Ableitung von Nutzeranforderungen“ benötigt werden, konnte so zum Beispiel der im Modell

dargestellte, gerichtete Pfeil von der Beschreibung des Nutzungskontextes zu den Nutzungsanforderungen nachgewiesen werden.

Die Liste der Aktivitäten und Artefakte ist nahezu vollständig und wurde auch sprachlich durch die Experten überarbeitet. So wurden zum Beispiel die Begriffe „Bedürfnisse“ und „Benutzeranforderungen“ durch die aktuellen, im Arbeitskreis für Qualitätsstandards verwendeten, Begriffe „Erfordernisse“ beziehungsweise „Nutzungsanforderungen“ geändert.

Anhand der durchgeführten Bewertung von Aktivitäten und Artefakten (siehe Anhang F - Tabellen F.3 und F.4) lässt sich erkennen, dass deren Relevanz durch die Experten als „neutral bis sehr bedeutend“ eingeschätzt wird. Lediglich zwei Aktivitäten und zwei Artefakte wurden als „weniger bedeutend bis unbedeutend“ bewertet. Um zu dieser Einschätzung zu gelangen, wurden die fünf Einheiten der Skala mit Werten von vier („sehr bedeutend“) bis null („unbedeutend“) belegt und anhand der Bewertungen der Experten der Mittelwert gebildet. Die Aktivitäten „*AKT.UE.26 - Dokumentieren von Begründungen, Faktoren und Gewichtungen der Mensch-System Belange*“ und „*AKT.UE.38 - Gestalten der Systeminfrastruktur*“ wurden dabei als „weniger bedeutend“ gekennzeichnet. Dies liegt vor allem daran, dass die Systeminfrastruktur vielmehr dem Software Engineering zugeordnet wird und weniger dem Usability Engineering. Bei den Artefakten wurden die „*ARF.UE.15 - Informationen über das Nutzungserlebnis*“, sowie die „*ARF.UE.29 - Gesundheitsberichte (medizinisch)*“ als „weniger bedeutend“ markiert. Des Weiteren konnten auch neue (ergänzende) Aktivitäten und Artefakte festgestellt und in das Modell aufgenommen werden. Diese sind gelistet in Tabelle 4.3.

Referenz	UE-Phase	Aktivität / Artefakt
<i>Aktivitäten:</i>		
AKT.UE.E1	VBN	Definieren von Zielen einer benutzerorientierten Gestaltung
AKT.UE.E2	FNA	Abgleichen von Nutzungsanforderungen zu Unternehmensanforderungen
AKT.UE.E3	BGL	Festlegen des Evaluationsgegenstandes und überprüfen, ob dieser zuverlässig messbar ist
AKT.UE.E4	BGL	Definieren der Zielsetzung einer Evaluation
<i>Artefakte:</i>		
ARF.UE.E1	EGL	Technische Rahmenbedingungen

Tabelle 4.3.: Ergänzende Aktivitäten und Artefakte durch die Expertenbefragung

In Kapitel 3.4 dieser Arbeit wurden acht Thesen in Bezug auf das Modell formuliert. Mit Hilfe der Fragen 18 bis 27 sollten diese validiert, beziehungsweise falsifiziert werden:

These 1 Die vier Phasen der ISO 9241-210 eignen sich auf Grund ihrer Abstraktheit als Rahmenwerk für ein ganzheitliches Integrationsmodell.

Lifecycle-Modelle des Software Engineering beginnen früher als das Usability Engineering und werden später beendet. So wird zum Beispiel bereits beim organisatorischen Bedarf für IT-Unterstützung angefangen und mit der „*Beerdigung des Systems*“ aufgehört. Zudem werden diese Phase von den Experten teilweise eher als Aktivitäten gesehen, die in mehr oder weniger großer Ausprägung über den gesamten Entwicklungszeitraum durchgeführt werden. Da das in dieser Arbeit entwickelte Integrationsmodell um die Phasen „Organisatorische Vorbereitungen“ und „System erfüllt festgelegte Anforderungen“ erweitert wurde, kann die These zwar nicht vollständig validiert, aber auch nicht falsifiziert werden.

These 2 Aktivitäten und Artefakte stehen in Wechselwirkung zueinander und bedingen sich gegenseitig (Artefaktlebenszyklus).

In einem Projekt wird faktisch immer mit den Ergebnissen einer Vorphase gearbeitet. Sollte diese nicht stattgefunden haben, werden die Ergebnisse dieser Vorphase antizipiert. So kann der durch das Modell in der ISO 9241-210 dargestellte logische Fluss bestätigt werden. Durch Fragen 9 und 10 bzw. 14 und 15 können die Wechselwirkungen zudem erhärtet werden. So werden durch die Experten die Ergebnisse einer Phase zugleich auch als notwendige Informationen einer darauf folgenden Phase beschrieben. Die These wurde somit belegt.

These 3 Die Analyse des Nutzungskontextes ist ausschlaggebend für das Verständnis des Problemraums und die Qualität der Lösung.

„*Eine Lösung ist nur so gut, wie sie ein Problem löst*“, lautet die Aussage eines Experten. Wird das Problem jedoch nicht verstanden, so kann es nicht gelöst werden. Das beste Verständnis eines Problems ist das zu befriedigende Erfordernis. „Analyse des Nutzungskontext“ wird dabei interpretiert als „*Analysieren, welche Erfordernisse im Nutzungskontext stecken*“. Um den Nutzungskontext zu analysieren, bedarf es vorher einer präzisen Beschreibung dessen. So können die aus den Nutzungskontextdaten erhaltenen Erfordernisse zur Spezifikation von Nutzungsanforderungen verwendet werden, wovon letztendlich die Qualität der Lösung abhängt. Ohne Kenntnis des Problems kann es im günstigsten Fall per Zufall zu sehr guten Lösungen kommen. Diese sind jedoch durch den Mangel an Systematik in deren Entwicklung mit einem hohen Risiko verbunden.

These 4 Eine gemeinsame Spezifikation von Anforderungen eignet sich als Ausgangsbasis für den weiteren Gestaltungs- und Entwicklungsprozess.

Die meisten Anforderungsspezifikationen unterscheiden nicht systematisch zwischen „Stakeholder-Anforderungen“ und „Systemanforderungen“, so dass die „Stakeholder-Anforderungen“ (aus den Erfordernissen der Benutzer) ignoriert werden. Eine gemeinsame Spezifikation von Anforderungen würde daher laut Experten *„absolut“* dazu beitragen, den weiteren Gestaltungs- und Entwicklungsprozess zu verbessern. Ein gemeinsames Dokument würde zu einer höheren Verbindlichkeit in der Umsetzung von Usability Anforderungen in der Organisation führen. Des Weiteren können der Mehraufwand und Interpretationsspielräume verringert werden, da Widersprüche und Inkonsistenzen vermieden werden. Idealerweise können auch weitere Dokumente des Usability Engineering (zum Beispiel die UI Spezifikation) mit Dokumenten des Software Engineering integriert werden, da neben der Anforderungsspezifikation auch Lösungsspezifikationen als Basis zur Umsetzung verwendet werden.

These 5 Eine Evaluation findet nicht nur am Ende jeder Iteration eines Gestaltungsprozesses statt, sondern sollte in jeder Phase durchgeführt werden.

Bereits bei einer Frage zu den Wechselwirkungen zwischen den Phasen, formulierte ein Experte die Aussage, dass die Evaluation (Validierung) *„über allen anderen schwebt“*. So sollten bereits Kontextdaten, Erfordernisse und Anforderungen durch Benutzer validiert werden. Wird bereits bei der Analyse am Budget gespart, je mehr Budget wird während der Evaluation benötigt, da gegebenenfalls Anforderungen und daran anschließende Interaktionsspezifikationen *„korrigiert“* werden müssen. Es sollte unterschieden werden zwischen einer „Validierung“ der Zwischenergebnisse mit den Benutzer und einer „Evaluation“ der spezifischen Lösungsidee.

These 6 Evaluationstechniken benötigen Modelle/Artefakte aller zuvor durchlaufenen Phasen als Bewertungsgrundlage.

Diese These trifft laut Experten absolut zu. Ohne ein entsprechendes Aufgabenmodell könnte beispielsweise die Technik des „Cognitive Walkthrough“ nicht durchgeführt werden. Alles andere wäre wie *„einen Nutzer durch einen Raum rennen lassen und gucken, wo er überall gegen rennt“*. Wäre dieses Problem behoben, könnte immer noch keine Aussage darüber getroffen werden, was der Mensch in diesem Raum hätte tun können. Eine Evaluation ohne eine Beschreibung des Nutzungskontextes und ohne Anforderungsspezifikation wäre lediglich *„ein guter erster Schritt“*, um formale Designmängel zu identifizieren.

These 7 Usability Engineering und Software Engineering besitzen keine identischen Aktivitäten oder Artefakte.

In der Einschätzung der Experten aus der Praxis wird vermutet, dass grundsätzlich 20 bis 30 Prozent identische Aktivitäten aus unterschiedlichen Perspektiven durchgeführt werden. Durch das Integrationsmodell konnte nachgewiesen werden, dass hauptsächlich die Aktivität der Anforderungsermittlung von beiden Disziplinen durchgeführt wird. Bei einer Evaluation des Usability Engineering wird beispielsweise die Akzeptanz durch den Benutzer betrachtet, während der Schwerpunkt einer Evaluation des Software Engineering auf dem Testen der Funktionalität beruht. Wie im Modell abgebildet und durch die Experten bestätigt wurde, liegt der Fokus im Software Engineering auf der technischen Umsetzung und nicht auf der Nutzung.

These 8 Artefakte beider Disziplinen gewinnen durch eine Integration gegenseitig an Qualität.

Ein Informationsaustausch zwischen Usability Engineering und Software Engineering steigert die Richtigkeit (Validität) des zu entwickelnden Systems. Ohne diesen Informationsaustausch wird lediglich sichergestellt, dass „richtig entwickelt wird“ (Verifikation). Dabei ist es sogar zwingend notwendig, „Synchronisationspunkte“ zu haben, an denen nicht nur Informationen, sondern auch Arbeitsprodukte ausgetauscht werden. Dabei sollten Artefakte des UE im SE berücksichtigt werden, aber auch Artefakte des SE im UE.

Im Rahmen einer Integration könnten, laut Experten, die Beschreibung des Nutzungskontextes und die Nutzungsanforderungen den Prozess des Requirements Engineering (*RE*) komplettieren. Des Weiteren sollte der Nutzungskontext mit technischen Rahmenbedingungen abgeglichen werden, sowie Nutzungsanforderungen mit Business Anforderungen und nicht-funktionalen Anforderungen (bspw. Datenschutz, Performanz, etc.). Gestaltungslösungen müssten mit technischen Lösungsspezifikationen abgeglichen werden und auch die Usability Evaluation sollte mit System- & Integrationstests verknüpft werden.

4.4. Zusammenfassung

In diesem Kapitel wurde die Durchführung einer expertenbasierten Befragung zur Beurteilung der Konstruktvalidität betrachtet. Dabei sollten der Bezug des Integrationsmodells zur Praxis, die Vollständigkeit und Korrektheit der Aktivitäten & Artefakte, sowie die in Kapitel 3.4 auf Grundlage des Modells formulierten Thesen nachgewiesen werden. Auf Grund der begrenzten Zeit der Masterthesis wurden zunächst nur zwei Experten aus der Perspektive des Usability Engineering befragt. Für die Formulierung geeigneter Fragen wurde das Themengebiet operationalisiert, indem der Untersuchungsgegenstand dimensional und semantisch analysiert wurde. So ergaben sich die vier in der Befragung

zu betrachtenden Aspekte „Akzeptanz“, „Vollständigkeit“, „Abhängigkeiten“ und „Thesen“ mit ihren dazugehörigen Dimensionen. Insgesamt konnten 27 Fragegegenstände erarbeitet und die dazugehörigen Fragen abgeleitet werden.

Anhand der Aussagen der Experten kann die Akzeptanz von Usability in Praxis auf strategischer und operativer Ebene nachgewiesen werden und wird von den Experten abhängig vom Kontext vorwiegend als „hoch“ eingeschätzt. Bei der Verwendung systematischer Ansätze einer benutzerorientierten Gestaltung sahen die Experten jedoch deutliche Defizite. Zwar wird der Benutzer in den Unternehmen betrachtet, doch geschieht dies eher willkürlich und nicht anhand eines systematischen Vorgehens. Ein Grund dafür ist der Mangel an Integration zwischen Usability Engineering und Software Engineering. So können geeignete Integrationsmodelle zu einer Verbesserung im Entwicklungsprozess führen.

Des Weiteren lässt sich auch die Vollständigkeit und Korrektheit der Aktivitäten, Artefakte und deren Abhängigkeiten bestätigen. Durch die Experten wurden lediglich zwei Aktivitäten und zwei Artefakte als „weniger bedeutend“ bewertet (siehe Abschnitt 4.3: AKT.UE.26, AKT.UE.38, ARF.UE.15 und ARF.UE.29). Da diese in der Summe der Expertenmeinungen jedoch nicht „unbedeutend“ sind, werden sie im Modell belassen. Auf der anderen Seite werden vier Aktivitäten und ein Artefakt im Modell anhand der Expertenaussagen ergänzt (siehe Abschnitt 4.3 - Tabelle 4.3).

Von den acht Thesen auf Grundlage des Modells, werden sechs durch die Experten validiert. Thesen 1 und 7 werden zwar nicht falsifiziert, können aber auch nicht eindeutig belegt werden. Das bedeutet, dass das Rahmenwerk des Integrationsmodells anhand der Phasen der ISO 9241-210 in seiner Vollständigkeit erweitert werden müsste, sofern aus dem Software Engineering auch Aktivitäten des Management oder der Stilllegung eines Systems berücksichtigt werden sollen. These 7 zu gemeinsamen Aktivitäten oder Artefakten der Disziplinen scheiterte vermutlich auf Grund der fehlenden sprachlichen Präzision des Begriffes „identisch“. Unter dem Begriff wurden redundant durchgeführte Aktivitäten oder erstellte Artefakte verstanden. Die Anforderungsspezifikation wird von beiden Disziplinen jeweils aus einer anderen Perspektive und gegebenenfalls in einer anderen Dokumentationsform erstellt, wodurch diese beispielsweise als „ähnlich“ beziehungsweise „ergänzend“ bezeichnet wurde.

Abschließend wird daher anhand der Ergebnisse die Konstruktvalidität des Integrationsmodells aus Sicht des Usability Engineering als nachgewiesen betrachtet. Im nächsten Kapitel wird diese Arbeit zusammengefasst und kritisch gewürdigt, sowie ein Ausblick auf verbleibende Fragen und Themen gegeben.

5. Zusammenfassung, kritische Würdigung und Ausblick

In diesem Kapitel wird die Arbeit noch einmal zusammengefasst, bevor sie kritisch gewürdigt wird. Anschließend wird ein Ausblick auf anknüpfende Themen und offene Fragen gegeben.

5.1. Zusammenfassung

Die Gebrauchstauglichkeit, als entscheidende Qualitätseigenschaft eines Produktes, gewinnt sowohl auf der strategischen, als auch auf der operativen Ebene einer Organisation zunehmend an Aufmerksamkeit. Dabei zählen verschiedene Aspekte, wie Wettbewerbsvorteil, Effizienzsteigerung, Kundenbindung, etc.. Jedoch mangelt es in vielen Unternehmen an einem systematischen Vorgehen zur Herstellung der Gebrauchstauglichkeit. Eine mögliche Erklärung dafür liegt in der fehlenden Integration von Usability Engineering und Software Engineering.

In dieser Masterthesis wurde daher das Thema der Integration von Usability Engineering und Software Engineering betrachtet und ein ganzheitliches Integrationsmodell entwickelt, evaluiert und optimiert.

Dazu wurden zunächst in Kapitel 2 Grundlagen erläutert, die für das Verständnis dieser Arbeit notwendig sind. Usability Engineering und Software Engineering wurden dabei als Disziplinen vorgestellt und anhand ihrer Entstehung, Ziele und Vorgehensmodelle näher beschrieben.

Der Fokus des Usability Engineering liegt auf der systematischen Bereitstellung von Vorgehensmodellen, Methoden und Techniken zur Gewährleistung der Gebrauchstauglichkeit bei der Entwicklung von Produkten. Dabei sollten die Benutzer frühzeitig in den Entwicklungsprozess eingebunden werden, um den Kontext, die Erfordernisse und die daraus resultierenden Anforderungen zu ermitteln. Diese werden als Basis zur kontinuierlichen Beurteilung von Lösungen benötigt.

Im Software Engineering liegt der Fokus auf der systematischen Bereitstellung von Prinzipien, Methoden, Techniken und Werkzeugen zur Entwicklung, zur Qualitätssi-

cherung, sowie zum Management großer Softwaresysteme. Besonders der Aspekt der Qualitätssicherung nimmt eine zentrale Rolle in aktuellen Vorgehensmodellen ein. Ausgehend von den zahlreichen, diskutierten Integrationsansätzen lässt sich nach Nebe (2009) eine Integration auf drei unterschiedlichen Abstraktionsebenen ansetzen: Standards, Modelle, Operativer Prozess. Eine vollständige Integration sollte Aspekte aus allen drei Ebenen berücksichtigen. Des Weiteren können Integrationsansätze in vier Kategorien eingeordnet werden. Ansätze der ersten Kategorie haben den Schwerpunkt auf Aspekten der konkreten Durchführung und definieren Aktivitäten, Artefakte, sowie Verknüpfungen zu bereits bestehenden Aktivitäten des Software Engineering. Bei Ansätzen der zweiten Kategorie wurde „der Weg des geringsten Widerstandes“ gewählt, indem gemeinsame Spezifikationen oder Artefakte betrachtet oder Artefakte einer Disziplin um Notationen gemäß der anderen Disziplin erweitert werden. Die Definition von Modellen und Prozessen als Vorgabe für die Entwicklung, stellt den Schwerpunkt der dritten Kategorie dar. Ansätze der vierten Kategorie sind abstrakter oder generischer Art und beinhalten allgemeine Rahmenbedingungen zur Berücksichtigung in einer Integration. Der in dieser Arbeit diskutierte Ansatz betrachtet die Standards des Usability Engineering und Software Engineering und wird der vierten Kategorie zugeordnet. Ein Überblick über die Standards ISO 9241-210, ISO/IEC 12207, ISO/TS 18152 und ISO/IEC 15504 wurde daher ebenfalls gegeben.

Im Rahmen eines studiumsbegleitenden Projektes konnte im Vorfeld der Masterthesis bereits an einem auf den Standards basierenden Integrationsmodells gearbeitet werden. Dieses ist Inhalt von Kapitel 3. Dabei wurden anhand der Aussagen in den Standards Aktivitäten und Artefakte identifiziert und in einem ganzheitlichen Modell abgebildet. Auf Grund der generischen Formulierung der Phasen der ISO 9241-210 und deren Bezug zu den Prinzipien (Analyse, Design, Entwicklung, Evaluation) des Software Engineering, wurden diese als Rahmenwerk des Modells verwendet.

Mit der Veröffentlichung der ISO 9241-210 als Nachfolger der DIN EN ISO 13407 wurde das Modell zur Wahrung der Aktualität überarbeitet. Dabei konnte die Anzahl der zuvor ermittelten Aktivitäten und Artefakte nahezu verdoppelt werden. Um diese zu bestätigen wurde eine intrinsische Validierung, in Form eines Abgleiches auf Basis der Standards ISO/TS 18152 und ISO/IEC 15504, durchgeführt. Zirka 70 Prozent des Modells konnten dabei belegt werden. Redundante Aktivitäten zwischen den Disziplinen wurden nicht festgestellt.

Wesentliche Erkenntnisse bis zu diesem Schritt waren die Identifikation der ergänzenden Disziplin des Marketing und der Marktforschung, sowie die Existenz eines Artefaktlebenszyklus und den dadurch entstehenden Wechselwirkungen zwischen den Phasen des Modells.

Um die unbestätigten 30 Prozent der Aktivitäten und Artefakte zu überprüfen, war eine weitere Validierung notwendig. Diese wurde auf Grund der begrenzten Bearbeitungszeit der Arbeit auf die Perspektive des Usability Engineering eingeschränkt. Anhand von Konformitäts- und Rahmenanforderungen nach Nebe (2009) konnte so eine extrinsische Validierung durchgeführt werden. Dazu wurde das Modell mit den formulierten Anforderungen, die auf dem Konsens mehrerer UE Experten beruhen, abgeglichen. Auch dabei konnten nicht alle Aktivitäten und Artefakte nachgewiesen werden. Andererseits wurde das Modell um 13 Aktivitäten erweitert. Abschließend konnten anhand des entwickelten Modells acht Thesen abgeleitet werden, die Rahmenanforderungen und Erkenntnisse am Modell beschreiben.

Um die Konstruktvalidität des Integrationsmodells zu überprüfen, wurde eine Expertenbefragung durchgeführt. Diese wurde in Kapitel 4 beschrieben. Ziel dabei war es den Bezug des Modells zur Praxis, die Vollständigkeit und Korrektheit der Aktivitäten und Artefakte, sowie die in Kapitel 3 formulierten Thesen nachzuweisen. Dazu wurde zunächst das Themengebiet, anhand einer dimensional und semantischen Analyse, operationalisiert.

Anhand der Aussagen der Experten konnte die Relevanz von Usability als hoch eingeschätzt werden. Jedoch sahen die Experten deutliche Defizite bei der systematischen Durchführung benutzerorientierter Gestaltungsansätze. Des Weiteren konnte auch die Vollständigkeit und Korrektheit der Aktivitäten, Artefakte und deren Abhängigkeiten belegt werden. So wurden vier neue Aktivitäten und ein neues Artefakt im Modell ergänzt. Von den acht Thesen konnten sechs validiert werden. Die zwei verbleibenden konnten zwar nicht eindeutig belegt, aber auch nicht falsifiziert werden. In Bezug auf das Rahmenwerk des Integrationsmodells bedeutet das, dass dieses erweitert werden müsste, sofern aus dem Software Engineering auch Aktivitäten des Management oder der Stilllegung eines Systems berücksichtigt werden sollen. Bei Gemeinsamkeiten wurden in dieser Arbeit differenziert zwischen beispielsweise „identischen“, „ähnlichen“ oder „ergänzenden“ Aktivitäten beziehungsweise Artefakten. Die Formulierung der These 7 zur Existenz „identischer Aktivitäten oder Artefakte“ muss daher anhand der fehlenden sprachlichen Präzision des Begriffes „identisch“ überarbeitet werden. Das Modell kann abschließend als konstruktvalide betrachtet werden.

5.2. Kritische Würdigung

Diese Arbeit hatte zum Ziel, auf Abstraktionsebene von Standards eine detaillierte Basis in Form eines strukturellen Modells der Aktivitäten und Artefakte zu schaffen, um

so mögliche Anknüpfungspunkte der Disziplinen zu identifizieren.

Das entstandene Modell, dargestellt in den Abbildungen in Anhang G, beinhaltet die aus den Standards ISO 9241-210 und ISO/IEC 12207 ermittelten Aktivitäten und Artefakte. Diese konnten in Anlehnung an die ISO 9241-210 mittels der Technik des Affinitätsdiagramms in die sechs Themen „*Organisatorische Vorbereitungen*“, „*Verstehen und Beschreiben des Nutzungskontext*“, „*Festlegen von Nutzungsanforderungen*“, „*Entwerfen von Gestaltungslösungen*“, „*Beurteilen von Gestaltungslösungen*“ und „*System erfüllt festgelegte Anforderungen*“ eingeordnet werden. Die Wahl dieser Themen, die in dieser Arbeit als „Phasen“ bezeichnet wurden, konnte nicht eindeutig validiert werden. Die Experten sind der Auffassung, dass der Umfang dieser Phasen nicht ausreicht, um auch die Aktivitäten des Software Engineering adäquat abzudecken. So bleiben beispielsweise zu Beginn die Einigungsprozesse mit dem Kunden, sowie zum Ende die Stilllegung des Systems, unberücksichtigt. Da der Fokus des Modells eindeutig auf den Entwicklungsprozess gelegt wurde, lassen sich aber die im Software Engineering verwendeten Phasen der Analyse, des Designs, der Entwicklung und des Testens mit dem Modell abbilden. Zu betonen sei laut Experten jedoch, dass es sich nicht um Phasen handelt, sondern vielmehr um Aktivitäten, die aus einer Detaillierung durch weitere Unteraktivitäten bestehen und welche im Projekt eher versetzt parallel als sequentiell ablaufen. Begründet wird dies damit, dass in der Abbildung im Standard ISO 9241-210 kein Prozessmodell beschrieben wird, sondern lediglich Abhängigkeiten (*Dependencies*) dargestellt werden.

Im Rahmen einer intrinsischen Validierung konnten die Aktivitäten und Artefakte im Modell erweitert und ein Großteil der Aktivitäten und Artefakte durch den Abgleich bestätigt werden. Bei dieser Art der Validierung handelt es sich laut eines Experten um einen „guten, ersten Schritt“. Da die beiden Standards ISO/TS 18152 und ISO/IEC 15504 jedoch die ISO 9241-210 bzw. DIN EN ISO 13407 und die ISO/IEC 12207 als Referenzmodell verwenden, kann lediglich die Vollständigkeit der identifizierten Aktivitäten nachgewiesen werden.

Da die Konformitäts- und Rahmenanforderungen nach Nebe (2009) mit Experten des Usability Engineering erarbeitet wurden, konnten neben der Vollständigkeit auch Abhängigkeiten der Aktivitäten untereinander und der Bezug des Modells zur Praxis betrachtet werden. Auf der einen Seite ließen sich zahlreiche Anforderungen auf Aktivitäten und auf das Modell als Ganzes abbilden. Auf der anderen Seite konnten aus den Anforderungen einige weitere Aktivitäten im Modell ergänzt werden.

Anhand der durchgeführten Befragung konnten nahezu alle Aktivitäten und Artefakte durch die Aussagen der Experten bestätigt werden. Inwieweit diese induktiv erarbeiteten Aussagen auf persönlichen Meinungen der Experten beruhen oder der allgemein-

gültigen Wahrheit entsprechen, kann nie vollständig sichergestellt werden. Der Nachweis der Konstruktvalidität wird daher lediglich vermutet. Des Weiteren basiert die vermutete Validität auf der Perspektive des Usability Engineering. Die Validität aus Perspektive des Software Engineering konnte auf Grund der begrenzten Zeit dieser Arbeit nicht betrachtet werden und ist demnach nicht gewährleistet.

Eine wesentliche Erkenntnis anhand des Modells ist der Lebenszyklus der Artefakte. So werden Artefakte nicht zum Selbstzweck erstellt, sondern dienen als Ausgangsbasis für darauf folgende Aktivitäten. Während Aktivitäten jeweils einer gewissen Phase beziehungsweise einer übergeordneten Aktivität zugehören, werden Artefakte phasenübergreifend verwendet und verknüpfen diese. Daraus entstehen logische Wechselwirkungen. Dementsprechend sollte eine Evaluation nicht nur zum Abschluss einer Iteration durchgeführt werden, sondern eine Validierung der Ergebnisse sollte bereits während der einzelnen Phasen stattfinden. Um gewisse Evaluationstechniken, beispielsweise „Cognitive Walkthrough“, „Think Aloud“ oder „Usability Testing“, durchzuführen, bedarf es sowohl einer Beschreibung des Nutzungskontextes, inklusive der darin enthaltenen Modelle, als auch einer Anforderungsspezifikation, welche die Erfordernisse der Benutzer formuliert. Daraus abgeleitet ist es von essentieller Bedeutung nicht nur vereinzelte Aktivitäten oder Artefakte des Usability Engineering mit dem Software Engineering zu integrieren, sondern sämtliche Aktivitäten und Artefakte beider Disziplinen zu berücksichtigen. Des Weiteren konnte im Hinblick auf die Integration der im Entwicklungsprozess beteiligten Disziplinen, Marketing und Marktforschung als weitere sehr wichtige Perspektive ermittelt und einige potentielle Aktivitäten im Modell verortet werden. So könnten Marketinginformationen die Analyse des Nutzungskontextes erweitern oder zur Validierung verwendet werden. Die Marktforschung könnte zudem nützliche Informationen zu den Benutzern des Systems, wie zum Beispiel deren Fähigkeiten, Vorlieben, Wünsche, etc., erarbeiten, um Benutzermodelle reichhaltiger zu gestalten. Auch Kenntnisse über eventuelle, vorherige Produkte, können bei der Gestaltung des zu entwickelnden Systems beitragen.

Gemäß des Ziels dieser Arbeit wurden Anknüpfungspunkte, von einem der Experten auch als „Synchronisationspunkte“ bezeichnet, zwischen den Disziplinen identifiziert. Dabei sollten neben den Informationen, auch Arbeitsprodukte ausgetauscht werden, so dass Artefakte des UE im SE und umgekehrt berücksichtigt werden.

Im Software Engineering wird der Nutzungskontext nicht betrachtet, während dieser für das Usability Engineering essentiell ist. In einem gemeinsamen Entwicklungsprozess sollte daher zuerst die Analyse des Nutzungskontextes durchgeführt werden, so dass der Kontext des zu entwickelnden Produktes dementsprechend abgegrenzt werden kann.

Ausgehend von denen im Nutzungskontext ermittelten Erfordernissen der Benutzer werden im Usability Engineering Nutzungsanforderungen spezifiziert. Der Schwerpunkt einer Anforderungsspezifikation im Software Engineering liegt hingegen auf den Business Anforderungen und den nicht-funktionalen Anforderungen, wie zum Beispiel Datenschutz, Performanz, etc.. Beide Disziplinen führen demnach die gleiche Aktivität durch, nämlich die Spezifikation von Anforderungen. Auf Grund des unterschiedlichen Fokus der Anforderungen wurde diese gemeinsame Aktivität jedoch als „nicht identisch“ beschrieben. Notwendig wäre es also, sämtliche Anforderungen in einem gemeinsamen Dokument zu erfassen und diese aus Perspektive beider Disziplinen gegeneinander abzuwägen beziehungsweise Zielkonflikte aufzulösen.

Des Weiteren sollten Ergebnisse der Gestaltungslösungen, zum Beispiel die UI Spezifikation, mit technischen Lösungsspezifikationen abgeglichen werden, um die angemessene Kommunikation zwischen Schnittstelle und System zu gewährleisten.

Auch die Usability Evaluation und System- & Integrationstests sollten gleichermaßen berücksichtigt werden, damit Probleme auf sämtlichen Ebenen des Systems identifiziert werden können.

Die gewonnenen Erkenntnisse und das erstellte Modell sind für eine theoretische Integration der Disziplinen des UE und SE sicherlich hilfreich. So konnten die durchzuführenden Aktivitäten und erstellten Artefakte abgebildet und Empfehlungen zur Integration formuliert werden. Um jedoch eine Motivation für entwickelnde Unternehmen herzustellen, müsste das Integrationsmodell auf Ebene der Unternehmensprozesse angepasst werden.

Auf Grund der gewonnenen Erkenntnisse im Projekt kann das Vorgehen im Nachhinein als „gut“ bezeichnet werden. Die Konstruktvalidität des Integrationsmodells müsste jedoch aus Perspektive des Software Engineering sichergestellt werden, um eine abschließende Aussage zur Bedeutung dieses Modells im Unternehmensprozess formulieren zu können.

Einen Ausblick, auf an diese Arbeit anknüpfende Themen und Aktivitäten, wird im nächsten Abschnitt gegeben.

5.3. Ausblick

Im Verlauf dieser Arbeit entstanden weitere interessante Fragestellungen und Aufgaben, die im zeitlichen Rahmen der Arbeit nicht berücksichtigt werden konnten und sich für anschließende Tätigkeiten anbieten.

- Um einen Bezug des Integrationsmodells zur Praxis herzustellen und so dessen Anwendung zu ermöglichen, müsste das Modell auf die Prozessebene abgebildet werden. So könnte ein Werkzeug, in Form eines Projektmanagementtools, auf Basis des Modells entwickelt werden. Jeder Aktivität müsste eine Rolle zugewiesen werden, so dass sich jede Person eine rollenspezifische Übersicht, der von ihm durchzuführenden Aktivitäten, anschauen könnte. Informationen und Artefakte könnten dann im System abgelegt und zu entsprechenden Personen kommuniziert werden, damit ein Austausch beziehungsweise eine Synchronisation stattfindet. Des Weiteren könnte eine Sicht auf Abhängigkeiten zwischen Aktivitäten und Artefakte gegeben werden, anhand derer betrachtet werden kann, welche Aktivitäten abgeschlossen werden müssen, damit mit einer anderen Aktivität begonnen werden kann.
- Bereits in Nebe (2009) ist die Notwendigkeit der Entwicklung eines Werkzeuges formuliert, welches die Durchführung der Bewertung von Vorgehensmodellen und Entwicklungsprozessen unterstützt beziehungsweise simplifiziert. Anhand dieses Integrationsmodells als Bewertungsgrundlage könnte ein entsprechendes Werkzeug entwickelt werden. Dieses könnte bei der Erfassung der Daten und der Analyse der Ergebnisse unterstützen, daraus auf die Reife des Vorgehensmodells oder den Entwicklungsprozess schließen, Lücken im Prozess ermitteln und zuverlässige Vorschläge zur Optimierung geben.
- Um Anforderungen des Usability Engineering und Software Engineering abzugleichen und einen gemeinsamen Konsens zu erstellen, könnte ein Werkzeug entwickelt werden, mit Hilfe dessen Anforderungen dokumentiert und in einer allgemein verständlichen Form repräsentiert werden. Des Weiteren könnten spezifisch der Rollen in einem Entwicklungsprozess verschiedene Sichten auf die Anforderungsspezifikation ermöglicht werden.
- Die bisherige Repräsentation des Modells in Form einer zweidimensionalen Abbildung und Aufteilung in vier Teilabbildungen ist sehr komplex und eventuell schwer zu erfassen. Daher wäre es sinnvoll eine weitere Möglichkeit der Repräsentation, zum Beispiel in Form einer semantischen Modellierung, zu finden. Diese sollte die Mehrdimensionalität des Modells unterstützen, Querverweise ermöglichen und interaktiv erfassbar sein.

So könnte eine andere Form der Repräsentation zum Beispiel auch als Datenbasis für die bereits angesprochenen Werkzeuge dienen.

- Die Gebrauchstauglichkeit eines Produktes sollte eine Selbstverständlichkeit darstellen. So ist die Kommunikation des Usability Engineering und deren Verortung in sämtlichen Phasen eines Prozesses essentiell.

Eine Vision besteht dem entsprechend in der Entwicklung eines eigenständigen Vorgehensmodells, welches die Disziplinen des Usability Engineering, Software Engineering und weitere Disziplinen, beispielsweise User Experience Design, User Interface Design, Marketing, vereint und somit eine zeitgemäße Entwicklung von Software für den Menschen ermöglicht.

Da in der heutigen Informationsgesellschaft besonders das Internet und demnach verteilte Anwendungen eine große Rolle spielen, bei deren Erstellung viele kleinere bis mittlere Unternehmen (5-30 Personen) beteiligt sind, sollte ein Hauptaugenmerk auch auf agile Vorgehensweisen gelegt werden.

- Bei der Identifikation der Aktivitäten im vorliegenden Modell lag der Schwerpunkt auf dem Entwicklungsprozess. So könnten ebenfalls die Sub-Prozesse zum Management etc. betrachtet und das Modell dahingehend erweitert werden. Dies könnte zu einer Steigerung der Effizienz und Effektivität in der Anwendbarkeit des Modells führen.
- Prozesse, die bereits jetzt eine hohe messbare Reife (High Maturity) aufweisen, könnten zur Evaluation des Modells begutachtet werden. So könnte festgestellt werden, wie sich die Reife der Prozesse nach Erweiterung durch das Modell verändert hat.
- Bei der Validierung des bisherigen Integrationsmodells wurde als induktives Verfahren eine Expertenbefragung durchgeführt. Es wurde sich bewusst für dieses wahrheitserweiternde Vorgehen entschieden, um auch neues Wissen anhand der Expertenaussagen zu erhalten und im Modell abzubilden. Des Weiteren müsste ebenfalls die Validität aus Perspektive des Software Engineering sichergestellt werden. Zudem könnte ein zusätzliches deduktives Verfahren angewandt werden, um den Wahrheitsgehalt des Integrationsmodells zu beurteilen.

Die vorliegende Masterthesis betrachtete die Thematik einer Integration von Usability Engineering und Software Engineering in einem ganzheitlichen Integrationsmodell, welches auf der Basis von Aktivitäten und Artefakte etablierter Standards der Disziplinen entwickelt, optimiert, sowie validiert wurde.

Die Betrachtung von anderen Integrationsansätzen und die Aussagen der Experten im Gespräch erlauben die Schlussfolgerung, dass eine Integration der Disziplinen in der

Praxis erwünscht ist. Bislang existiert jedoch kein allgemeiner Konsens, so dass das in dieser Arbeit erstellte Integrationsmodell als mögliches Rahmenwerk zur Integration verwendet werden kann.

Die Integration der Disziplinen bleibt weiterhin ein interessantes Thema, welches zu verfolgen sich lohnt, und wird in der Zukunft maßgeblich zur Qualität eines Produktes beitragen.

Abbildungsverzeichnis

2.1. Usability Engineering Lifecycle	17
2.2. Scenario-based Framework	19
2.3. Goal-Directed Design Process	20
2.4. V-Modell	22
2.5. Rational Unified Process	24
2.6. Scrum	24
2.7. Bausteine des Marketing-Konzeptes	26
2.8. Abstraktionsebenen von Integrationsansätzen	30
2.9. Kategorien von UE Standards	38
2.10. Benutzerorientierte Gestaltung interaktiver Systeme	39
2.11. Struktur der ISO/IEC 12207	45
3.1. Extrahierte Aktivität auf einem Notizzettel	52
3.2. Grundmodell der sechs Oberthemen (Phasen)	53
3.3. Affinitätsdiagramm der Aktivitäten	54
3.4. Beispiel für ein/e digital repräsentierte/s Aktivität und Artefakt	56
3.5. Gesamtansicht des Integrationsmodells	57
3.6. Ausschnitt des Teilmodells der Aktivitäten	58
3.7. Ausschnitt des Teilmodells des UE	58
3.8. Ausschnitt des Teilmodells des SE	59
3.9. Ausschnitt des Teilmodells der Artefakte	60
3.10. Vereinfachtes Modell der Wechselwirkungen zwischen Phasen	61
3.11. Beispiele für Aktivitäten und Artefakte des Marketing und der Marktforschung	62
3.12. Ausschnitt des Teilmodells der Aktivitäten	73
4.1. Bewertungsvorlage Fragen F.1 - F.4	82
4.2. Bewertungsvorlage Fragen F.7, F.13 & F.17	83
G.1. Teilmodell von Aktivitäten des UE & SE - Übersicht	180
G.2. Teilmodell von Aktivitäten des UE & SE - Teil I	181
G.3. Teilmodell von Aktivitäten des UE & SE - Teil II	182
G.4. Teilmodell von Aktivitäten des UE & SE - Teil III	183
G.5. Teilmodell von Aktivitäten des UE & SE - Teil IV	184
G.6. Teilmodell von Aktivitäten und Artefakte des UE - Übersicht	185
G.7. Teilmodell von Aktivitäten und Artefakte des UE - Teil I	186
G.8. Teilmodell von Aktivitäten und Artefakte des UE - Teil II	187
G.9. Teilmodell von Aktivitäten und Artefakte des UE - Teil III	188
G.10. Teilmodell von Aktivitäten und Artefakte des UE - Teil IV	189

G.11.Teilmodell von Aktivitäten und Artefakte des SE - Übersicht	190
G.12.Teilmodell von Aktivitäten und Artefakte des SE - Teil I	191
G.13.Teilmodell von Aktivitäten und Artefakte des SE - Teil II	192
G.14.Teilmodell von Aktivitäten und Artefakte des SE - Teil III	193
G.15.Teilmodell von Aktivitäten und Artefakte des SE - Teil IV	194
G.16.Teilmodell von Artefakten der UE & SE - Übersicht	195
G.17.Teilmodell von Artefakten des UE & SE - Teil I	196
G.18.Teilmodell von Artefakten des UE & SE - Teil I	197
G.19.Teilmodell von Artefakten des UE & SE - Teil I	198
G.20.Teilmodell von Artefakten des UE & SE - Teil I	199
G.21.Modelle des UE	200

Tabellenverzeichnis

2.1. Reifestufen nach Teil 2 ISO/IEC 15504	47
3.1. Prozentuale Verwendung der Standards	51
3.2. Phasenübergreifende Abhängigkeiten der Artefakte	60
3.3. Prozentuale Verwendung der Standards zur Prozessbewertung	63
3.4. Ausschnitt der Base Practices ISO/TS 18152	64
3.5. Ausschnitt der Base Practices ISO/IEC 15504	65
3.6. Ausschnitt der Gemeinsamkeiten von Aktivitäten	66
3.7. Referenzen Aktivitäten UE (Ausschnitt)	67
3.8. Aktivitäten anhand der Konformitäts- und Rahmenanforderungen	72
4.1. Befragungsplan	81
4.2. Einschätzung zur Relevanz von Usability und HCD	84
4.3. Ergänzende Aktivitäten und Artefakte	86
A.1. Referenzen Aktivitäten UE	118
A.2. Referenzen Aktivitäten SE	123
A.3. Referenzen Aktivitäten MM	124
A.4. Referenzen Artefakte UE	127
A.5. Referenzen Artefakte SE	129
A.6. Referenzen Artefakte MM	130
B.1. Base Practices ISO/TS 18152	135
B.2. Base Practices ISO/IEC 15504	142
C.1. Artefakt-Lebenszyklus UE	144
C.2. Artefakt-Lebenszyklus SE	145
D.1. Gemeinsamkeiten Aktivitäten	151
D.2. Gemeinsamkeiten Artefakte	152
E.1. Konformitäts- und Rahmenanforderungen	162
F.1. Dimensionale Analyse des Untersuchungsgegenstandes	167
F.2. Gewichtung der Phasen	168
F.3. Gewichtung der Aktivitäten	172
F.4. Gewichtung der Artefakte	174
F.5. Aussagen der Experten	178

Literaturverzeichnis

Balzert 2001

BALZERT, Helmut: *Lehrbuch der Software-Technik - Software-Entwicklung*. 2. überarbeitete Auflage. Spektrum-Akademischer Verlag, Heidelberg, 2001. – ISBN 978-3-8274-0480-0

Balzert 2008

BALZERT, Helmut: *Lehrbuch der Software-Technik - Softwaremanagement*. 2. Auflage. Spektrum-Akademischer Verlag, Heidelberg, 2008. – ISBN 978-3-8274-1161-7

Beck 1999

BECK, Kent: *Extreme Programming Explained: Embrace Change*. Addison-Wesley, Harlow, England, 1999. – ISBN 978-0-201-61541-5

Benyon et al. 2005

BENYON, David; TURNER, Phil; TURNER, Susan: *Designing Interactive Systems: People, Activities, Contexts, Technologies*. Addison-Wesley, Harlow, England, 2005. – ISBN 978-0-321-11629-1

Bevan 2001

BEVAN, Nigel: *International Standards for HCI and Usability*. In: *International Journal of Human Computer Studies*. Volume 55, No. 4. Academic Press, Boston, 2001, S. 533–552

Beyer & Holtzblatt 1997

BEYER, Hugh; HOLTZBLATT, Karen: *Contextual Design - Defining Customer-Centered Systems*. Morgan Kaufmann Publishers, San Francisco, USA, 1997. – ISBN 978-1-55860-411-7

Boehm 1988

BOEHM, Barry W.: *A spiral model of software development and enhancement*. In: *Computer*, IEEE, Volume 21, Mai, 1988, S. 61–72

Bortz & Döring 2006

BORTZ, Jürgen; DÖRING, Nicola: *Forschungsmethoden und Evaluation: für Human- und Sozialwissenschaftler*. 4. überarbeitete Auflage. Springer Verlag, Berlin, 2006. – ISBN 978-3-5403-3305-0

BSI 2010

BSI (Hrsg.): *What is a standard?* Version: 2010. <http://www.bsigroup.com/en/Standards-and-Publications/About-standards/What-is-a-standard/>, Sichtung: 13.06.2010

Buxton & Randell 1969

BUXTON, John; RANDELL, Brian: *Software Engineering Techniques: Report of a conference sponsored by the NATO Science Committee, Rom.* Scientific Affairs Division, Brüssel, 1969

Cockburn 2000

COCKBURN, Alistair: *Writing Effective Use Cases.* Addison-Wesley, Harlow, England, 2000. – ISBN 978-0-201-70225-5

Cockburn 2004

COCKBURN, Alistair: *Crystal Clear: A Human-Powered Methodology for Small Teams.* Addison-Wesley, Harlow, England, 2004. – ISBN 978-0-201-69947-0

Constantine et al. 2003

CONSTANTINE, Larry; BIDDLE, Robert; NOBLE, James: *Usage-Centered Design and Software Engineering: Models for Integration.* In: (ICSE 2003), S. 106–113

Constantine & Lockwood 1999

CONSTANTINE, Larry L.; LOCKWOOD, Lucy A. D.: *Software for Use: A Practical Guide to the Models and Methods of Usage-Centered Design.* Addison-Wesley, Harlow, England, 1999. – ISBN 978-0-201-92478-7

Cooper et al. 2007

COOPER, Alan; REIMANN, Robert; CRONIN, David: *About Face 3.0 - The Essentials of Interaction Design.* 3. Auflage. John Wiley & Sons Ltd, Chichester, England, 2007. – ISBN 978-0-470-08411-3

DIN EN ISO 13407 1999

Norm DIN EN ISO 13407 : 1999. *Human-centred design processes for interactive systems*

Dix et al. 2004

DIX, Alan; FINLAY, Janet; ABOWD, Gregory D.; BEALE, Russell: *Human-Computer Interaction.* 3. Auflage. Pearson Education Limited, Harlow, England, 2004. – ISBN 978-0-13-046109-4

Düchting et al. 2007

DÜCHTING, Markus; ZIMMERMANN, Dirk; NEBE, Karsten: *Incorporating User Centered Requirement Engineering into Agile Software Development.* In: *Proceedings of the HCII 2007*, Volume 4550. Springer, Berlin. Februar, 2007

Edmonds 1974

EDMONDS, Ernest A.: *A process for the development of software for non-technical users as an adaptive system*, General Systems XIX, 1974, S. 215–218

Faulkner 2000

FAULKNER, Xristine: *Usability Engineering.* Palgrave Macmillan, New York, 2000. – ISBN 978-0-33377-321-5

Ferre 2003

FERRE, Xavier: *Integration of Usability Techniques into the Software Development Process*. In: (ICSE 2003), S. 28–35

Garrett 2002

GARRETT, Jesse J.: *The Elements of User Experience*. New Riders, Berkeley, USA, 2002. – ISBN 978-0-7357-1202-7

Good et al. 1986

GOOD, Michael; SPINE, Thomas M.; WHITESIDE, John; GEORGE, Peter: *User-Derived Impact Analysis as a Tool for Usability Engineering*. In: *CHI' 86: Proceedings of the SIGCHI conference on Human factors in computing systems*. New York, 1986, S. 241–246

Göransson et al. 2003

GÖRANSSON, Bengt; LIF, Magnus; GULLIKSEN, Jan: *Usability Design - Extending Rational Unified Process with a New Discipline*. In: *Interactive Systems. Design, Specification and Verification*, Volume 2844, pp. 15. Springer, Berlin, Heidelberg. Mai, 2003

Hakiel 1997

HAKIEL, Simon: *Delivering ease of use*. In: *Computing & Control Engineering Journal*, IEEE, Volume 8, No. 2, April, 1997, S. 81–87

Hassenzahl et al. 2008

HASSENZAHN, Marc; BURMESTER, Michael; KOLLER, Franz: *Der User Experience (UX) auf der Spur: Zum Einsatz von www.attrakdiff.de*. In: BRAU, Henning (Hrsg.); DIEFENBACH, Sarah (Hrsg.); HASSENZAHN, Marc (Hrsg.); KOLLER, Franz (Hrsg.); PEISSNER, Matthias (Hrsg.); RÖSE, Kerstin (Hrsg.): *Usability Professionals 2008: Berichtband des sechsten Workshops des German Chapters der Usability Professionals Association e.V.*, German Chapter der Usability Professionals' Association e.V., 2008. – ISBN 978-3-816-77769-4, S. 78–82

Hennigs & Zimmermann 2009

HENNIGS, Lennart (Hrsg.); ZIMMERMANN, Dirk (Hrsg.): *User Centered Design im Unternehmen - Im Spannungsfeld zwischen Theorie und Praxis*. Version: 2009. http://www.lennartgroetzbach.de/downloads/WUD09_UCD_im_Unternehmen.pdf, Sichtung: 02.06.2010

Herczeg 2009

HERCZEG, Michael: *Software-Ergonomie: Interaktive Medien - Theorien, Modelle und Kriterien für gebrauchstaugliche interaktive Computersysteme*. 3. erw. Auflage. Oldenbourg Wissenschaftsverlag, München, 2009. – ISBN 978-3-486-58725-8

Hix & Hartson 1993

HIX, Deborah; HARTSON, H. R.: *Developing User Interfaces: Ensuring Usability Through Product & Process*. John Wiley & Sons Ltd, New York, USA, 1993. – ISBN 978-0-471-57813-0

Holtzblatt et al. 2005

HOLTZBLATT, Karen; WENDELL, Jessamyn B.; WOOD, Shelley: *Rapid Contextual Design - A How-To Guide to Key Techniques for User-Centered Design*. Morgan Kaufmann Publishers, San Francisco, USA, 2005. – ISBN 978-0-12-354051-5

IABG 1995

IABG (Hrsg.): *V-Modell*. Version: 1995. <http://v-modell.iabg.de>, Sichtung: 10.06.2010

IABG 1997

IABG (Hrsg.): *V-Modell '97*. Version: 1997. <http://v-modell.iabg.de/dmdocuments/V-Modell-97-doc-englisch.zip>, Sichtung: 10.06.2010

IABG 2009

IABG (Hrsg.): *V-Modell XT V1.3*. Version: 2009. <http://v-modell.iabg.de/dmdocuments/V-Modell-XT-Gesamt-Deutsch-V1.3.pdf>, Sichtung: 10.06.2010

ICSE 2003

ICSE (Hrsg.): *Bridging the Gaps Between Software Engineering and Human-Computer Interaction*. ICSE'03 International Conference on Software Engineering, Portland, Oregon, 2003

ISO 2010a

ISO (Hrsg.): *ISO/PAS Publicly available specification*. Version: 2010. http://www.iso.org/iso/standards_development/processes_and_procedures/deliverables/iso_pas_deliverable.htm, Sichtung: 16.06.2010

ISO 2010b

ISO (Hrsg.): *ISO/TS Technical specification*. Version: 2010. http://www.iso.org/iso/standards_development/processes_and_procedures/deliverables/iso_ts_deliverable.htm, Sichtung: 16.06.2010

ISO 9241-11 1998

Norm ISO 9241-11 : 1998. *Ergonomic requirements for office work with visual display terminals - Part 11: Guidance on usability*

ISO 9241-110 2006

Norm ISO 9241-110 : 2006. *Ergonomics of human-system interaction - Part 110: Dialogue principles*

ISO 9241-210 2010

Norm ISO 9241-210 : 2010. *Ergonomics of human-system interaction - Part 210: Human-centred design for interactive systems*

ISO/IEC 12207 1996

Norm ISO/IEC 12207 : 1996. *Standard for Information Technology - Software life cycle processes (Part 0-2)*

ISO/IEC 12207 2002

Norm ISO/IEC 12207 : 2002. *Standard for Information Technology - Software life cycle processes - Amendment 1*

ISO/IEC 15504 2006

Norm ISO/IEC 15504 : 2006. *Information technology - Software process assessment (Part 1-5)*

ISO/PAS 18152 2003

Vornorm ISO/PAS 18152 : 2003. *Ergonomics of human-system interaction - Specification for the process assessment of human-system issues*

ISO/TS 18152 2010

Vornorm ISO/TS 18152 : 2010. *Ergonomics of human-system interaction - Specification for the process assessment of human-system issues*

Juristo et al. 2003

JURISTO, Natalia; LOPEZ, Marta; MORENO, Ana M.; SÁNCHEZ, M. I.: *Improving software usability through architectural patterns*. In: **(ICSE 2003)**, S. 12–19

Karat & Karat 2003

KARAT, John; KARAT, Clare-Marie: *The evolution of user-centered focus in the human-computer interaction field*. In: *IBM Systems Journal*, Vol. 42, NO 4, 2003, S. 532–541

Kawakita 1982

KAWAKITA, Jiro: *The Original KJ Method*. KJ Method Headquarters, Kawakita Research Institute, Tokyo, 1982

Kotler et al. 2006

KOTLER, Philip; ARMSTRONG, Gary; SAUNDERS, John; WONG, Veronica: *Grundlagen des Marketing*. 4. akt. Auflage. Pearson Education Deutschland GmbH, München, 2006. – ISBN 978–3–8273–7176–8

Kotonya & Sommerville 1998

KOTONYA, Gerald; SOMMERVILLE, Ian: *Requirements Engineering: Processes and Techniques*. John Wiley & Sons Ltd, Chichester, England, 1998. – ISBN 978–0–471–97208–2

Kromrey 2009

KROMREY, Helmut: *Empirische Sozialforschung*. 12. neu bearbeitete Auflage. UTB GmbH, Stuttgart, 2009. – ISBN 978–3–8252–1040–3

Kruchten 2003

KRUCHTEN, Philippe: *The Rational Unified Process: An Introduction*. 3. Auflage. Addison-Wesley, Harlow, England, 2003. – ISBN 978–0–321–19770–2

Leffingwell & Widrig 2003

LEFFINGWELL, Dean; WIDRIG, Don: *Managing Software Requirements - A Use Case Approach*. 2. Auflage. Addison-Wesley, Harlow, England, 2003. – ISBN 978–0–321–12247–6

Lewis & Rieman 1993

LEWIS, Clayton (Hrsg.); RIEMAN, John (Hrsg.): *Task-Centered User Interface Design*. Version: 1993. <http://hcibib.org/tcuid/tcuid.pdf>, Sichtung: 09.04.2010

Likert 1932

LIKERT, Rensis: *A Technique for the Measurement of Attitudes*. In: *Archives of Psychology*, Volume 140, 1932, S. 1–55

de Luca 1998

LUCA, Jeff de (Hrsg.): *The Latest FDD Processes V1.3*. Version: 1998. <http://www.nebulon.com/articles/fdd/download/fddprocessesA4.pdf>, Sichtung: 10.06.2010

Mayhew 1999

MAYHEW, Deborah J.: *The Usability Engineering Lifecycle - A Practitioner's Handbook for User Interface Design*. Morgan Kaufmann Publishers, San Francisco, USA, 1999. – ISBN 978–1–55860–561–9

McCracken & Jackson 1982

MCCRACKEN, Daniel D.; JACKSON, Michael A.: *Life cycle concept considered harmful*. In: *ACM SIGSOFT Software Engineering Notes*, ACM, Volume 7, No. 2, April, 1982, S. 29–32

Metzker & Reiterer 2002

METZKER, Eduard; REITERER, Harald: *Evidence-Based Usability Engineering*. In: KOLSKI, Christophe (Hrsg.); VANDERDONCKT, Jean (Hrsg.): *Computer-Aided Design of User Interfaces III: Proceedings of the Fourth International Conference on Computer-Aided Design of User Interfaces, Valenciennes, France*. Kluwer Academics Publishing, Dordrecht, 2002. – ISBN 978–1–4020–0643–2, S. 323–336

Nebe 2009

NEBE, Karsten: *Integration von Usability Engineering und Software Engineering - Konformitäts- und Rahmenanforderungen zur Bewertung und Definition von Softwareentwicklungsprozessen*. Dissertation. Shaker Verlag GmbH, Aachen, 2009. – ISBN 978–3–8322–8074–1

Nebe et al. 2007

NEBE, Karsten; DÜCHTING, Markus; ZIMMERMANN, Dirk: *Integration von User Centred Design Aktivitäten in Agile Softwareentwicklung*. In: RÖSE, Kerstin (Hrsg.); BRAU, Henning (Hrsg.): *Usability Professionals 2007: Berichtband des fünften Workshops des German Chapters der Usability Professionals Association e.V.*, German Chapter der Usability Professionals' Association e.V., 2007. – ISBN 978–3–8167–7414–3, S. 1–6

Nebe et al. 2006

NEBE, Karsten; GRÖTZBACH, Lennart; HARTWIG, Ronald: *Integrating User Centred Design in a Product Development Lifecycle Process: A Case Study*. In: ARABNIA, Hamid R. (Hrsg.); REZA, Hassan (Hrsg.): *Software Engineering Research and Practice*, CSREA Press, 2006. – ISBN 1–932415–91–2, S. 695–701

Niedziella 2007

NIEDZIELLA, Wolfgang: *Wie funktioniert Normung?: Eine Einführung in die nationale (DIN/DKE), europäische (CENELEC) und internationale (IEC) elektrotechnische Normung*. 2. überarbeitete Auflage. VDE Verlag, Berlin, 2007. – ISBN 978-3-80073-006-3

Nielsen 1992

NIELSEN, Jakob: *The Usability Engineering Life Cycle*. In: *Computer*, IEEE, Volume 25, No. 3, März, 1992, S. 12–22

de Piotrowski & Tauber 2009

PIOTROWSKI, Arkadiusz M. F.; TAUBER, Michael J.: *Benutzerprofile von Menschen mit Beeinträchtigungen/Fähigkeiten*. In: WANDKE, Hartmut (Hrsg.); KAIN, Saskia (Hrsg.); STRUVE, Doreen (Hrsg.): *Mensch & Computer 2009: Grenzenlos frei!?* Oldenbourg Verlag, München, 2009. – ISBN 978-3-486-59222-1, S. 33–42

Preece et al. 2007

PREECE, Jenny; ROGERS, Yvonne; SHARP, Helen: *Interaction Design: Beyond Human-Computer Interaction*. 2. Auflage. John Wiley & Sons Ltd, Chichester, England, 2007. – ISBN 978-0-470-01866-8

Preece et al. 1994

PREECE, Jenny; ROGERS, Yvonne; SHARP, Helen; BENYON, David; HOLLAND, Simon; CAREY, Tom: *Human-Computer Interaction*. Addison-Wesley Longman, Harlow, England, 1994. – ISBN 978-0-201-62769-5

Rosson 1999

ROSSON, Mary B.: *Integrating development of task and object models*. In: *Communications of the ACM*, ACM, Volume 42, No. 1, Januar, 1999, S. 49–56

Rosson & Carroll 2002

ROSSON, Mary B.; CARROLL, John M.: *Usability Engineering: Scenario-Based Development of Human-Computer Interaction*. Morgan Kaufmann Publishers, San Francisco, USA, 2002. – ISBN 978-1-55860-712-5

Royce 1970

ROYCE, Winston W.: *Managing the development of large software systems*. In: *Proceedings of IEEE Wescon*, IEEE, August, 1970, S. 1–9

Schaffer 2004

SCHAFER, Eric: *Institutionalization of Usability: A Step-by-Step Guide*. Addison-Wesley, Boston, USA, 2004. – ISBN 978-0-321-17934-0

Seffah et al. 2005a

SEFFAH, Ahmed; DESMARAIS, Michel C.; METZKER, Eduard: *HCI, Usability and Software Engineering Integration: Present and Future*. In: (Seffah et al. 2005c), S. 37–57

Seffah et al. 2005b

SEFFAH, Ahmed; GULLIKSEN, Jan; DESMARAIS, Michel: *An Introduction to Human-Centered Software Engineering*. In: (Seffah et al. 2005c), S. 3–14

Seffah et al. 2005c

SEFFAH, Ahmed (Hrsg.); GULLIKSEN, Jan (Hrsg.); DESMARAIS, Michel C. (Hrsg.): *Human-Centered Software Engineering: Integrating Usability in the Software Development Lifecycle*. Springer Netherlands, Dordrecht, 2005. – ISBN 978–1–4020–4027–6

Seffah & Metzker 2004

SEFFAH, Ahmed; METZKER, Eduard: *The obstacles and myths of usability and software engineering*. In: *Communications of the ACM*, ACM, Volume 47, No. 12, December, 2004, S. 71–76

Shneiderman & Plaisant 2009

SHNEIDERMAN, Ben; PLAISANT, Catherine: *Designing the User Interface - Strategies for Effective Human-Computer Interaction*. 5. überarbeitete Auflage. Addison-Wesley, Harlow, England, 2009. – ISBN 978–0–321–60148–3

Sisson 1986

SISSON, Norwood: *Dialogue management reference model*. In: ACM (Hrsg.): *ACM SIGCHI Bulletin, New York, USA, October, 1986, Volume 18, Issue 2*, 1986, S. 34–35

Sousa et al. 2005

SOUSA, Kenia; FURTADO, Elizabeth; MENDONÇA, Hildeberto: *UPi - A Software Development Process Aiming at Usability, Productivity and Integration*. In: *CLIHC '05: Proceedings of the 2005 Latin American Conference on Human-Computer Interaction, Cuernavaca, Mexiko*. ACM Press, New York, 2005, S. 76–87

Staudacher & Langenbacher 2008

STAUDACHER, Holger; LANGENBACHER, Tobias: *Agiles Projektmanagement mit Scrum*. Grin Verlag, München, 2008. – ISBN 978–3–6401–3396–3

Stewart & Travis 2003

STEWART, Tom; TRAVIS, David: *Guidelines, Standards, and Style Guides*. In: JACKO, Julie A. (Hrsg.); SEARS, Andrew (Hrsg.): *The Human-Computer Interaction Handbook - Fundamentals, Evolving Technologies and Emerging Applications*. Lawrence Erlbaum Associates Inc., Mahwah, New Jersey, USA, 2003. – ISBN 978–0–805–84468–9, S. 991–1005

Sutcliffe 2005

SUTCLIFFE, Allistair G.: *Convergence or Competition between Software Engineering and Human Computer Interaction*. In: (Seffah et al. 2005c), S. 71–84

SWEBOK 2004

SWEBOK: *Guide to the Software Engineering Body of Knowledge by IEEE Computer Society Professional Practices Committee*. 2004

Takeuchi & Nonaka 1982

TAKEUCHI, Hirotaka; NONAKA, Ikujiro: *The new new product development game*. In: *Harvard Business Review*, Volume 64, No. 1, 1982, S. 137–146

Voigt 2010

VOIGT, Kai-Ingo (Hrsg.): *Gabler Wirtschaftslexikon, Stichwort: Norm*. Version: 2010. <http://wirtschaftslexikon.gabler.de/Archiv/54979/norm-v4.html>, Sichtung: 03.03.2010

Winter 2005

WINTER, Mario: *Methodische objektorientierte Softwareentwicklung*. dpunkt.verlag GmbH, Heidelberg, 2005. – ISBN 978–3–89864–273–6

Zuser et al. 2004

ZUSER, Wolfgang; GRECHENIG, Thomas; KÖHLE, Monika: *Software Engineering. Mit UML und dem Unified Process*. 2. überarbeitete Auflage. Pearson Education Deutschland GmbH, München, 2004. – ISBN 978–3–8273–7090–7

Anhang

A. Referenzen

Sammlung der aus den Standards ISO 9241-210, DIN EN ISO 13407 & ISO/IEC 12207 identifizierten und in Kapitel 3 beschriebenen Aktivitäten und Artefakten.

Erläuterungen zu Tabellen A.1 bis A.6:

- Referenz: Eindeutige Bezeichnung einer Aktivität bzw. eines Artefakts
- UE-Phase: Einordnung einer Aktivität bzw. eines Artefakts in eine Phase des UE: „*Organisatorische Vorbereitungen*“, „*Verstehen und Beschreiben des Nutzungskontext*“ (VBN), „*Festlegen von Nutzungsanforderungen*“ (FNA), „*Entwerfen von Gestaltungslösungen*“ (EGL), „*Beurteilen von Gestaltungslösungen*“ (BGL), „*System erfüllt festgelegte Anforderungen*“ (SAF)
- Aktivität: Beschreibung der Aktivität
- Artefakt: Beschreibung des Artefakts
- Quelle: Verortung der Aktivität bzw. des Artefakts in der jeweiligen Norm
- Bezug: Verortung der Aktivität bzw. des Artefakts in den Standards zur Prozessbewertung ISO/TS 18152 und ISO/IEC 15504

A.1. Aktivitäten

Referenz	UE-Phase	Aktivität	Quelle ISO 9241-210	Bezug ISO 13407	Bezug ISO/PAS 18152
AKT.UE.1		Festlegen der Notwendigkeit einer benutzer-orientierten Gestaltung	9241-210:6.1	13407:7.1	HS.2.1.BP1 HS.2.1.BP4
AKT.UE.2		Identifizieren von adäquaten Methoden und Ressourcen	9241-210:5.3.a	-	HS.2.6.BP4
AKT.UE.3		Definieren von Prozeduren zur Integration von Aktivitäten und Artefakten mit anderen Systementwicklungsaktivitäten	9241-210:5.3.b	-	HS.2.4.BP3
AKT.UE.4		Identifizieren der für die benutzerorientierten Gestaltungsaktivitäten zuständigen Personen und Organisationen, sowie deren Fähigkeiten und Perspektiven	9241-210:5.3.c	-	HS.1.3.BP4 HS.2.2.BP5 HS.2.4.BP2 HS.4.1.BP1 HS.4.1.BP2 HS.4.1.BP3 HS.4.2.BP3
AKT.UE.5		Entwickeln von effektiven Prozeduren zur Etablierung von Rückmeldungen und Kommunikation über benutzerorientierte Gestaltungsaktivitäten	9241-210:5.3.d	13407:7.3.1.c 13407:7.3.1.g	HS.1.3.BP3 HS.2.7.BP1 HS.2.7.BP3 HS.2.8.BP3
AKT.UE.6		Entwickeln von effektiven Methoden zur Dokumentation der Erkenntnisse	9241-210:5.3.d	13407:7.5.7	HS.2.8.BP4
AKT.UE.7		Einigen über angemessene Meilensteine für benutzerorientierte Aktivitäten, integriert in den übergeordneten Design- und Entwicklungsprozess	9241-210:5.3.e	-	HS.4.3.BP1
AKT.UE.8		Einigen über entsprechende Zeitskalen für Iterationen, den Gebrauch von Rückmeldungen und mögliche Gestaltungsänderungen, eingegliedert in den Projektplan	9241-210:5.3.f	13407:7.4.6	-
AKT.UE.9		Definieren einer Strategie und eines Plans zur Einbindung von Benutzer	-	-	HS.2.6.BP3
AKT.UE.10	VBN	Eingrenzen des Nutzungskontextes (Scope)	-	-	HS.3.1.BP1
AKT.UE.11	VBN	Identifizieren von Benutzern und anderen Stakeholdern	9241-210:6.2.2.a	-	HS.1.1.BP4
AKT.UE.12	VBN	Identifizieren der Charakteristiken von Benutzern und Benutzergruppen	9241-210:6.2.2.b	13407:7.2.1.a	HS.3.1.BP3

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität	Quelle ISO 9241-210	Bezug ISO 13407	Bezug ISO/PAS 18152
AKT.UE.13	VBN	Identifizieren der Aufgaben der Benutzer und damit verbundene Ziele	9241-210:6.2.2.c	13407:7.2.1.b	HS.1.1.BP3 HS.3.1.BP2
AKT.UE.14	VBN	Identifizieren der Umgebung(en) des Systems (organisatorisch, sozial, kulturell, physisch)	9241-210:6.2.2.d	13407:7.2.1.c	HS.1.1.BP1 HS.1.1.BP2 HS.2.7.BP5 HS.3.1.BP4 HS.3.1.BP5 HS.3.1.BP6
AKT.UE.15	VBN	Beeinflussen bzw. Abändern der Aufgaben und organisatorischen Rahmenbedingungen	-	-	HS.1.1.BP7
AKT.UE.16	VBN	Einschätzen von Auswirkungen auf die Gesellschaft und das Umfeld durch menschliche Fehler beim Gebrauch des Systems	-	-	HS.1.2.BP7 HS.1.4.BP6
AKT.UE.17	VBN	Einschätzen von Risiken für die physische und psychische Gesundheit und das Wohlbefinden der Benutzer selbst	-	-	HS.1.4.BP5
AKT.UE.18	FNA	Identifizieren der Erfordernisse (Needs) von Benutzern und anderen Stakeholdern	9241-210:6.3.2	-	HS.1.1.BP5 HS.1.2.BP5 HS.2.3.BP1
AKT.UE.19	FNA	Ableiten von Nutzungsanforderungen	9241-210:6.3.3	-	HS.3.2.BP2
AKT.UE.20	FNA	Ableiten von Anforderungen aus dem Nutzungskontext	9241-210:6.3.3.a	13407:7.3.1.b	HS.3.1.BP7
AKT.UE.21	FNA	Ableiten von Anforderungen aus den Erfordernissen (Needs) im Nutzungskontext	9241-210:6.3.3.b	-	-
AKT.UE.22	FNA	Ableiten von Anforderungen aus relevanten Kenntnissen über Ergonomie und Benutzungsschnittstellen, Standards und Richtlinien	9241-210:6.3.3.c	-	HS.1.2.BP6 HS.2.5.BP1 HS.2.8.BP2
AKT.UE.23	FNA	Aufstellen von messbaren Kriterien der Gebrauchstauglichkeit	9241-210:6.3.3.d	13407:7.3.2.d	HS.3.2.BP4
AKT.UE.24	FNA	Ableiten von organisatorischen Anforderungen, die den Benutzer direkt betreffen	9241-210:6.3.3.e	-	-
AKT.UE.25	FNA	Beseitigen von Zielkonflikten zwischen Nutzungsanforderungen	9241-210:6.3.4	-	HS.3.2.BP3

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität	Quelle ISO 9241-210	Bezug ISO 13407	Bezug ISO / PAS 18152
AKT.UE.26	FNA	Dokumentieren von Begründungen, Faktoren und Gewichtungen der Mensch-System Belange	9241-210:6.3.4	13407:7.3.1.j	-
AKT.UE.27	FNA	Sicherstellen der Qualität der Nutzungsanforderungsspezifikation (überprüfbar formuliert, verifiziert durch Stakeholder, konsistent, aktualisiert)	9241-210:6.3.5	13407:7.2.2.c 13407:7.3.2.e	-
AKT.UE.28	EGL	Gestalten von Benutzeraufgaben	9241-210:6.4.1.a 9241-210:6.4.2.2	13407:7.3.1.d 13407:7.3.1.e 13407:7.3.1.f	HS.3.1.BP2
AKT.UE.29	EGL	Gestalten von Benutzer-System-Interaktion unter Berücksichtigung von Nutzungserlebnis (UX)	9241-210:6.4.1.a 9241-210:6.4.2.2	-	HS.3.2.BP1 HS.3.3.BP3
AKT.UE.30	EGL	Berücksichtigen von Gestaltungsprinzipien (siehe ISO 9241-110: Aufgabenangemessenheit, Selbstbeschreibungsfähigkeit, Erwartungskonformität, Lernförderlichkeit, Steuerbarkeit, Fehlertoleranz, Individualisierbarkeit)	9241-210:6.4.2.1	13407:7.4.2	-
AKT.UE.31	EGL	Aufstellen von Gestaltungsentscheidungen auf konzeptueller Ebene	9241-210:6.4.2.2	-	-
AKT.UE.32	EGL	Identifizieren von Aufgaben und Teilaufgaben	9241-210:6.4.2.2	13407:7.3.1.d 13407:7.3.1.e	-
AKT.UE.33	EGL	Zuordnen von Aufgaben und Teilaufgaben zu Benutzern und anderen Systemkomponenten	9241-210:6.4.2.2	13407:7.3.1.f	HS.2.7.BP2 HS.3.3.BP1
AKT.UE.34	EGL	Identifizieren von benötigten Interaktionsobjekten	9241-210:6.4.2.2	-	-
AKT.UE.35	EGL	Identifizieren und Auswählen von angemessenen Dialogtechniken (siehe ISO 9241-12 bis ISO 9241-17)	9241-210:6.4.2.2	-	-
AKT.UE.36	EGL	Gestalten von Sequenzen der Interaktion	9241-210:6.4.2.2	-	-
AKT.UE.37	EGL	Gestalten der Informationsarchitektur der Benutzungsschnittstelle	9241-210:6.4.2.2	-	-
AKT.UE.38	EGL	Gestalten der Systeminfrastruktur	-	-	HS.1.3.BP1 HS.1.3.BP2
AKT.UE.39	EGL	Gestalten der Benutzungsschnittstelle unter Berücksichtigung von Nutzungserlebnis (UX)	9241-210:6.4.1.a 9241-210:6.4.2.3	13407:7.3.1.i	HS.3.3.BP3
AKT.UE.40	EGL	Berücksichtigen von relevanten Kenntnissen über Ergonomie und Benutzungsschnittstellen, Standards und Richtlinien	9241-210:6.4.2.3	13407:7.4.2	-

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität	Quelle ISO 9241-210	Bezug ISO 13407	Bezug ISO/PAS 18152
AKT.UE.41	EGL	Berücksichtigen von internen Gestaltungsrichtlinien für Benutzungsschnittstellen, Produktkenntnissen, Kenntnissen über die Benutzer und weiteren Aspekten des Nutzungskontextes (z.B. UX und Stereotypen)	9241-210:6.4.2.3	13407:7.4.2	HS.1.4.BP3
AKT.UE.42	EGL	Konkretisieren von Gestaltungslösungen (mittels Szenarien, Simulationen, Modellen, Prototypen)	9241-210:6.4.1.b 9241-210:6.4.3	13407:7.4.3	HS.1.2.BP1 HS.1.2.BP2 HS.1.2.BP3 HS.1.2.BP4 HS.3.3.BP2
AKT.UE.43	EGL	Modifizieren von Gestaltungslösungen auf Basis von benutzerorientierter Evaluation und Rückmeldungen	9241-210:6.4.1.c 9241-210:6.4.4	13407:7.4.4 13407:7.4.5	HS.1.4.BP4 HS.3.3.BP5
AKT.UE.44	EGL	Kommunizieren der Gestaltungslösung zu Verantwortlichen der Implementierung	9241-210:6.4.1.d 9241-210:6.4.5	13407:7.3.1.c	HS.3.3.BP4
AKT.UE.45	BGL	Sammeln von neuen Informationen über die Erfordernisse (Needs) der Benutzer	9241-210:6.5.1.a	-	HS.1.5.BP1
AKT.UE.46	BGL	Bereitstellen von Rückmeldungen über Stärken und Schwächen der Gestaltungslösung aus Sicht der Benutzer (Formative Evaluation)	9241-210:6.5.1.b	13407:7.5.3	HS.2.5.BP3 HS.3.4.BP5
AKT.UE.47	BGL	Beurteilen, ob die Benutzeranforderungen erreicht wurden (Konformitätsprüfung)	9241-210:6.5.1.c	13407:7.5.4	HS.1.3.BP5 HS.3.4.BP4
AKT.UE.48	BGL	Durchführen einer benutzerorientierten Evaluation	9241-210:6.5.2	-	HS.3.1.BP8 HS.3.2.BP5 HS.4.4.BP5
AKT.UE.49	BGL	Bereitstellen von Ressourcen	9241-210:6.5.2.a	-	-
AKT.UE.50	BGL	Planen einer benutzerorientierten Evaluation	9241-210:6.5.2.b	13407:7.5.2	HS.3.4.BP1 HS.3.4.BP2 HS.3.4.BP3 HS.4.4.BP1
AKT.UE.51	BGL	Durchführen einer umfassenden Prüfung, um Ergebnisse über das System als Ganzes zu erhalten	9241-210:6.5.2.c	-	HS.3.4.BP4
AKT.UE.52	BGL	Ergebnisse analysieren und erforderliche Anpassung der Lösungen entwickeln	9241-210:6.5.2.d	-	HS.1.3.BP6 HS.3.4.BP4 HS.4.4.BP4

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität	Quelle ISO 9241-210	Bezug ISO 13407	Bezug ISO / PAS 18152
AKT.UE.53	BGL	Kommunizieren der Ergebnisse	9241-210:6.5.2.e	-	-
AKT.UE.54	BGL	Testen mit Benutzern	9241-210:6.5.4	-	HS.3.4.BP4
AKT.UE.55	BGL	Beurteilen, ob Nutzungsanforderungen erreicht wurden	9241-210:6.5.4	13407:7.5.4	HS.3.4.BP4
AKT.UE.56	BGL	Durchführen einer Feldvalidierung	9241-210:6.5.4	13407:7.5.5	HS.3.4.BP4
AKT.UE.57	BGL	Durchführen einer inspektionsbasierten Prüfung und Bewertung (Expertentest)	9241-210:6.5.5	-	HS.2.5.BP2 HS.2.8.BP6 HS.3.4.BP4 HS.4.4.BP3
AKT.UE.58	BGL	Durchführen von Langzeitbeobachtungen	9241-210:6.5.6	13407:7.5.6	HS.3.4.BP4
AKT.UE.59	BGL	Sicherstellen, dass gewonnene Erkenntnisse und Daten angewandt werden bzw. Einfluss nehmen	-	-	HS.4.4.BP6

Tabelle A.1.: Referenzen der Aktivitäten des Usability Engineering

Referenz	UE-Phase	Aktivität	Quelle ISO/IEC 12207	Bezug ISO/IEC 15504
AKT.SE.1		Festlegen eines Vorgehensmodells	12207:5.3.1	ENG.1.BP1
AKT.SE.2		Migrieren eines Produktes	12207:5.5.5	ENG.1.1.BP5 ENG.1.2.BP5 ENG.2.BP6
AKT.SE.3		Stilllegen eines Produktes	12207:5.5.6	ENG.2.BP7
AKT.SE.4	VCN	Dokumentieren der Benutzer	12207:5.3.4.1.i 12207:5.3.5.4	-
AKT.SE.5	VCN	Aktualisieren der Dokumentation der Benutzer	12207:5.3.6.4 12207:5.3.7.3 12207:5.3.8.3 12207:5.3.9.2	-
AKT.SE.6	FNA	Analysieren des beabsichtigten Nutzens des Systems	12207:5.3.2	CUS.1.BP1
AKT.SE.7	FNA	Spezifizieren von Business Anforderungen	12207:5.3.2	CUS.3.BP1
AKT.SE.8	FNA	Spezifizieren von Funktionen und Fähigkeiten des Systems	12207:5.3.2	ENG.1.1.BP1 ENG.1.1.BP2
AKT.SE.9	FNA	Spezifizieren von Anforderungen bezüglich Sicherheit, Schutz und Ergonomie	12207:5.3.2 12207:5.3.4.1.d 12207:5.3.4.1.e 12207:5.3.4.1.f	ENG.1.1.BP2 ENG.1.2.BP1
AKT.SE.10	FNA	Spezifizieren von Benutzeranforderungen zu Ablauf und Ausführung	12207:5.3.4.1.j	-
AKT.SE.11	FNA	Spezifizieren von Arbeitsprozessen/-abläufen	12207:5.3.2	ENG.1.2.BP2
AKT.SE.12	FNA	Spezifizieren von organisatorischen Anforderungen	12207:5.3.2	-
AKT.SE.13	FNA	Spezifizieren von Anforderungen an Betrieb und Wartung	12207:5.3.2	ENG.2.BP1
AKT.SE.14	FNA	Spezifizieren von Benutzeranforderungen an Betrieb und Wartung	12207:5.3.4.1.k	-
AKT.SE.15	FNA	Spezifizieren der Schnittstellen	12207:5.3.2 12207:5.3.4.1.b	ENG.1.3.BP2
AKT.SE.16	FNA	Sicherstellen der Testbarkeit von Anforderungen	12207:5.3.2.2.c 12207:5.3.4.2.d	CUS.3.BP6 ENG.1.BP3 ENG.1.2.BP4

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität	Quelle ISO/IEC 12207	Bezug ISO/IEC 15504
AKT.SE.17	FNA	Sicherstellen der Nachvollziehbarkeit zu erfassten Bedürfnissen, Anforderungen und Entwürfen	12207:5.3.2.2.a 12207:5.3.3.2.a 12207:5.3.4.2.a 12207:5.3.5.6.a 12207:5.3.6.7.a 12207:5.3.7.5.a	ENG.1.1.BP2 ENG.1.1.BP7 ENG.1.3.BP5 ENG.1.4.BP4
AKT.SE.18	FNA	Sicherstellen der Konsistenz	12207:5.3.2.2.b 12207:5.3.3.2.b 12207:5.3.4.2.bc 12207:5.3.5.6.bc 12207:5.3.6.7.bc 12207:5.3.7.5.bc	ENG.1.2.BP8
AKT.SE.19	FNA	Spezifizieren von Benutzeranforderungen	12207:5.3.2	-
AKT.SE.20	FNA	Spezifizieren der Daten und Anforderungen an die Datenhaltung	12207:5.3.4.1.g	ENG.1.1.BP1
AKT.SE.21	FNA	Spezifizieren von Installations- und Akzeptanzanforderungen	12207:5.3.4.1.h	CUS.1.1.BP4
AKT.SE.22	FNA	Spezifizieren von Qualifizierungsanforderungen	12207:5.3.2 12207:5.3.4.1.c	CUS.3.BP3
AKT.SE.23	FNA	Evaluieren und validieren von Anforderungen	-	CUS.3.BP2 ENG.1.2.BP3
AKT.SE.24	FNA	Managen von Änderungen an Anforderungen	-	CUS.3.BP4 ENG.1.2.BP6
AKT.SE.25	FNA	Kommunizieren von Anforderungen	-	ENG.1.1.BP6 ENG.1.2.BP7
AKT.SE.26	EGL	Umwandeln der Anforderungen an die Software in eine top-level Softwarestruktur	12207:5.3.5.1	ENG.1.1.BP4 ENG.1.3.BP1
AKT.SE.27	EGL	Berücksichtigen von Designeinschränkungen	12207:5.3.2	-
AKT.SE.28	EGL	Durchführen von gemeinsamen Überprüfungen (unter den Entwicklern; Joint Reviews)	12207:5.3.5.7 12207:5.3.6.8 12207:5.3.8.6	SUP.6.BP1

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität	Quelle ISO/IEC 12207	Bezug ISO/IEC 15504
AKT.SE.29	EGL	Entwickeln von Modifikationen	12207:5.5.3	ENG.2.BP5
AKT.SE.30	EGL	Entwerfen einer top-level Systemarchitektur	12207:5.3.3	ENG.1.1.BP3
AKT.SE.31	EGL	Entwerfen eines detaillierten Softwaredesigns mit programmierbaren Einheiten	12207:5.3.6.1	ENG.1.3.BP4
AKT.SE.32	EGL	Entwickeln der Softwareeinheiten	12207:5.3.7.1	ENG.1.4.BP1
AKT.SE.33	EGL	Entwickeln von Testprozeduren und Testdaten	12207:5.3.7.1	ENG.1.4.BP2
AKT.SE.34	EGL	Entwerfen eines top-level Designs der Datenhaltung	12207:5.3.5.3	-
AKT.SE.35	EGL	Entwerfen eines detaillierten Designs der Datenhaltung	12207:5.3.6.3	-
AKT.SE.36	EGL	Entwerfen eines top-level Designs der internen und externen Schnittstellen	12207:5.3.5.2	ENG.1.3.BP2
AKT.SE.37	EGL	Entwerfen eines detaillierten Designs der Schnittstellen	12207:5.3.6.2	ENG.1.3.BP2
AKT.SE.38	EGL	Entwickeln von Tests für Qualifizierungsanforderungen	12207:5.3.8.4	-
AKT.SE.39	EGL	Spezifizieren von Funktionen und Fähigkeiten der Software	12207:5.3.4.1.a	ENG.1.3.BP1
AKT.SE.40	EGL	Integrieren von Softwareeinheiten	-	ENG.1.5.BP5
AKT.SE.41	EGL	Aggregieren von Systemeinheiten	-	ENG.1.7.BP3
AKT.SE.42	BGL	Planen des Ablaufs der Softwareintegration und definieren von Testanforderungen	12207:5.3.5.5	ENG.1.6.BP1 ENG.1.6.BP2
AKT.SE.43	BGL	Aktualisieren des Ablaufplans der Softwareintegration und der Testanforderungen	12207:5.3.6.6	ENG.1.6.BP1 ENG.1.6.BP2
AKT.SE.44	BGL	Planen von Tests der Softwareeinheiten und definieren von Testanforderungen	12207:5.3.6.5	ENG.1.5.BP2 ENG.1.5.BP3
AKT.SE.45	BGL	Aktualisieren des Ablaufplans der Tests der Softwareeinheiten und der Testanforderungen	12207:5.3.7.4	ENG.1.5.BP2 ENG.1.5.BP3
AKT.SE.46	BGL	Planen einer Strategie zur Systemintegration und definieren von Tests	-	ENG.1.7.BP1 ENG.1.7.BP2 ENG.1.7.BP4 ENG.1.7.BP6

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität	Quelle ISO/IEC 12207	Bezug ISO/IEC 15504
AKT.SE.47	BGL	Planen einer Strategie zur Wartung	-	ENG.2.BP2
AKT.SE.48	BGL	Aufzeichnen und bereitstellen von Unterstützungen der Benutzer beim Ausführen der Software	12207:5.4.4.1 12207:5.4.4.2 12207:5.4.4.3	CUS.4.1.BP1
AKT.SE.49	BGL	Evaluiere die Software und Testergebnisse	12207:5.3.7.5	ENG.1.6.BP3 ENG.1.6.BP4
AKT.SE.50	BGL	Evaluiere das Design, des Codes, der Tests, der Testergebnisse und der Benutzerdokumentation	12207:5.3.9.3	-
AKT.SE.51	BGL	Durchführen von Tests der Softwareeinheiten und Datenbasis gegenüber den Anforderungen	12207:5.3.7.2	ENG.1.4.BP3 ENG.1.5.BP4 ENG.1.5.BP6
AKT.SE.52	BGL	Planen der Evaluation des Produktes in der beabsichtigten Umgebung	12207:5.4.1.1	CUS.4.BP2
AKT.SE.53	BGL	Anlegen von Prozeduren für die Erfassung, Erkennung, Aufzeichnung und Unterstützung von Rückmeldungen und Problemen	12207:5.4.1.2	CUS.4.1.BP4 SUP.8.BP1
AKT.SE.54	BGL	Anlegen von Prozeduren für das Testen des Produktes in der beabsichtigten Umgebung	12207:5.4.1.3	CUS.4.BP2
AKT.SE.55	BGL	Durchführen der angelegten Prozeduren zur Evaluation in der beabsichtigten Umgebung	12207:5.4.2.1 12207:5.4.2.2 12207:5.4.3.1	CUS.4.1.BP2
AKT.SE.56	BGL	Durchführen der Wartung	12207:5.5.2 12207:5.5.3	CUS.4.1.BP5 ENG.2.BP3
AKT.SE.57	BGL	Bewerten von Modifikationen	12207:5.5.4	ENG.2.BP4
AKT.SE.58	BGL	Bewerten der Angemessenheit von Designstandards und verwendeten Methoden	12207:5.3.3.2.c	SUP.4.BP2
AKT.SE.59	BGL	Bewerten der Nachvollziehbarkeit und Konsistenz zu Anforderungen	12207:5.3.3.2.a 12207:5.3.3.2.b	ENG.1.1.BP7 SUP.5.BP2
AKT.SE.60	BGL	Bewerten der Umsetzbarkeit von Abläufen und Wartung	12207:5.3.2.2.e 12207:5.3.3.2.e 12207:5.3.4.2.f 12207:5.3.5.6.f 12207:5.3.6.7.f 12207:5.3.7.5.g	-

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität	Quelle ISO/IEC 12207	Bezug ISO/IEC 15504
AKT.SE.61	BGL	Bewerten der Umsetzbarkeit von Softwareelementen zur Ausführung der zugewiesenen Anforderungen	12207:5.3.3.2.d	ENG.1.3.BP3
AKT.SE.62	BGL	Bewerten der Umsetzbarkeit des Systemarchitekturdesigns	12207:5.3.2.2.d	-
AKT.SE.63	BGL	Planen der Integration der Einheiten in die Software	12207:5.3.8.1	ENG.1.5.BP1
AKT.SE.64	BGL	Integrieren der Softwareeinheiten und testen der Aggregationen	12207:5.3.8.2	ENG.1.7.BP8
AKT.SE.65	BGL	Integrieren der Software in das System und testen	12207:5.3.10.1	ENG.1.7.BP5 ENG.1.7.BP7
AKT.SE.66	BGL	Durchführen von Akzeptanztests und Softwaretests	12207:5.3.13.1	CUS.1.4.BP1 CUS.1.4.BP2
AKT.SE.67	BGL	Durchführen von Qualifizierungstests	12207:5.3.9.1	SUP.3.BP4 SUP.3.BP5
AKT.SE.68	BGL	Durchführen von Systemqualifizierungstests	12207:5.3.11.1	SUP.3.BP4 SUP.3.BP5
AKT.SE.69	BGL	Durchführen von Audits	12207:5.3.9.1 12207:5.3.11.3	SUP.7.BP1
AKT.SE.70	BGL	Reagieren auf Qualitäts-, Verifikations- und Validierungsergebnisse	-	SUP.3.BP6 SUP.3.BP7 SUP.4.BP3 SUP.5.BP3
AKT.SE.71	SAF	Planen und durchführen der Installation	12207:5.3.12.1 12207:5.3.13.2	CUS.2.BP4 CUS.3.BP5
AKT.SE.72	SAF	Ausliefern des Produktes	12207:5.3.13.2	CUS.2.BP4 CUS.3.BP5
AKT.SE.73	SAF	Gewährleisten von Training und Support	12207:5.3.13.3	CUS.4.2.BP1

Tabelle A.2.: Referenzen der Aktivitäten des Software Engineering

Referenz	UE-Phase	Aktivität	Quelle Kotler 2006
AKT.MM.1		Festlegen einer kundenorientierten Denkweise	S. 71f
AKT.MM.2	VBN	Suchen von Käufern	S. 38
AKT.MM.3	VBN	Erkennen der Bedürfnisse der Käufer	S. 38
AKT.MM.4	VBN	Festlegen von Marktsegmenten	S. 38f
AKT.MM.5	VBN	Analysieren des Konsumentenverhaltens	S. 310ff
AKT.MM.6	VBN	Untersuchen der makroökonomischen Umgebung	S. 147ff
AKT.MM.7	VBN	Analysieren der Märkte	S. 143f
AKT.MM.8	VBN	Analysieren der Konkurrenten	S. 144, 578ff
AKT.MM.9	VBN	Analysieren der Öffentlichkeit	S. 144ff
AKT.MM.10	VBN	Analysieren der organisationsbezogenen Einflussgrößen	S. 372ff
AKT.MM.11	VBN	Durchführen eines Customer Relationship-Managements (CRM)	S. 560ff
AKT.MM.12	EGL	Bereitstellen von Marketinginformationen	ISO 13407:7.4.2
AKT.MM.13	EGL	Berücksichtigen von Markenbildung und Werbung im Rahmen des Nutzungserlebnisses (UX)	ISO 9241-210:4.6
AKT.MM.14	BGL	Feststellen der Zielerreichung	S. 51, 71 537ff, 552ff
AKT.MM.15	SAF	Werben	S. 38
AKT.MM.16	SAF	Festlegen von Preisen	S. 38
AKT.MM.17	SAF	Lagern & Ausliefern	S. 38

Tabelle A.3.: Referenzen der Aktivitäten des Marketing & der Marktforschung

A.2. Artefakte

Referenz	UE-Phase	Artefakt	Quelle ISO 9241-210	Bezug ISO 13407	Bezug ISO / PAS 18152
ARF.UE.1		Plan der benutzerorientierten Gestaltung	9241-210:5.3	-	A.26:16
ARF.UE.2		Nachweis der Kommunikation	9241-210:5.3.d 9241-210:6.4.1.d	13407:Tab.C.1	A.26:87
ARF.UE.3		Ziele der benutzerorientierten Gestaltung	9241-210:3	13407:Tab.C.3	A.26:13
ARF.UE.4	VBN	Beschreibung des gegebenen Nutzungskontextes	9241-210:3 9241-210:6.2.2	13407:7.2.2 9241-11:5.3	A.26:110
ARF.UE.5	VBN	Beschreibung des festgelegten Nutzungskontextes für das Design	9241-210:3	-	A.26:110
ARF.UE.6	FNA	Beschreibung der Erfordernisse der Benutzer	9241-210:3	-	-
ARF.UE.7	FNA	Ziele der Gebrauchstauglichkeit	9241-210:6.5.4	-	A.26:12
ARF.UE.8	FNA	Messbare Kriterien (Gebrauchstauglichkeit, Zufriedenheit)	9241-210:6.5.4	13407:Tab.C.3	A.26:36
ARF.UE.9	FNA	Spezifikation der Nutzungsanforderungen	9241-210:3	-	A.26:52
ARF.UE.10	FNA	Informationen über Ergonomie	9241-210:6.3.3.c	-	-
ARF.UE.11	FNA	Informationen über Benutzungsschnittstellengestaltung	9241-210:6.3.3.c	-	-
ARF.UE.12	FNA	Standards / Normen	9241-210:6.3.3.c	13407:7.4.2	A.26:9
ARF.UE.13	FNA	Richtlinien (u.a. Styleguides des Unternehmens)	9241-210:6.3.3.c	-	A.26:116
ARF.UE.14	EGL	Spezifikation der Benutzerinteraktion (einschließlich Aufgaben und Teilaufgaben)	9241-210:3	-	-
ARF.UE.15	EGL	Informationen über das Nutzungserlebnis	9241-210:6.4.1	-	-
ARF.UE.16	EGL	Designprinzipien (u.a. ISO 9241-110)	9241-210:6.4.2.1	-	A.26:116
ARF.UE.17	EGL	Dialogtechniken (ISO 9241-12 bis ISO 9241-17)	9241-210:6.4.2.2	-	A.26:116
ARF.UE.18	EGL	Gestaltungsrichtlinien	9241-210:6.4.2.3	13407:7.4.2	-
ARF.UE.19	EGL	Produktkenntnisse	9241-210:6.4.2.3	13407:7.4.2	-
ARF.UE.20	EGL	Marketinginformationen (Vorlieben, Abneigungen, Wünsche)	9241-210:6.4.2.3	13407:7.4.2	A.26:46
ARF.UE.21	EGL	Spezifikation der Benutzungsschnittstelle	9241-210:3	-	-
ARF.UE.22	EGL	Umsetzung der Benutzungsschnittstelle in einem Prototyp	9241-210:3	13407:7.4	-

Weiter auf der nächsten Seite

Referenz	UE-Phase	Artefakt	Quelle ISO 9241-210	Bezug ISO 13407	Bezug ISO/PAS 18152
ARF.UE.23	BGL	Evaluationsplan	9241-210:6.5.2.b	13407:Tab.C.5	A.26:59
ARF.UE.24	BGL	Ergebnisse der Evaluation	9241-210:3	13407:7.5.7 13407:Tab.C.2 13407:Tab.C.3 13407:Tab.C.4	A.26:62
ARF.UE.25	BGL	Ergebnisse der Konformitätsprüfung	9241-210:3	-	A.26:41
ARF.UE.26	BGL	Ergebnisse der Langzeitbeobachtung (z.B. Rückmeldungen)	9241-210:3	-	-
ARF.UE.27	BGL	Rückmeldungen	9241-210:6.5.1	13407:Tab.C.4	A.26:31
ARF.UE.28	BGL	Leistungskriterien	9241-210:6.5.6	13407:7.5.6	-
ARF.UE.29	BGL	Gesundheitsberichte (medizinisch)	9241-210:6.5.6	13407:7.5.6	-
ARF.UE.30	BGL	Bericht über Zielerreichung	9241-210:6.5.1.c	13407:7.5.4	-
ARF.UE.31	BGL	Bericht über die Behandlung von Nichtkonformitäten	9241-210:6.5.1.c 9241-210:8.c	13407:Tab.C.5	-

Tabelle A.4.: Referenzen der Artefakte des Usability Engineering

Referenz	UE-Phase	Artefakt	Quelle ISO/IEC 12207	Bezug ISO/IEC 15504
ARF.SE.1		Beschreibung von Standards der Softwareentwicklung	12207-1:6.17	C.2:9
ARF.SE.2		Projektmanagementplan	12207-1:6.11	C.2:17
ARF.SE.3		Plan des Entwicklungsprozesses	12207-1:6.5	C.2:16
ARF.SE.4		Konzept/Bedarfsfeststellung	12207-1:4.3	C.2:44
ARF.SE.5		Softwareproduktbeschreibung	12207-1:4.3	C.2:13
ARF.SE.6		Akzeptanzstrategie	12207-1:4.3	C.2:68
ARF.SE.7	VBN	Geschäftslogik	12207-1:6.3	C.2:12
ARF.SE.8	VBN	Beschreibung der Benutzerdokumentation	12207-1:6.30	C.2:105
ARF.SE.9	FNA	Anforderungsbeschreibung	12207-1:6.22	C.2:52
ARF.SE.10	FNA	Systemanforderungsspezifikation	12207-1:6.26	C.2:53
ARF.SE.11	EGL	Test- bzw. Validierungsprozeduren	12207-1:6.28	C.2:64 C.2:65 C.2:66 C.2:67
ARF.SE.12	EGL	Aufzeichnung über ausführbaren Code	12207-1:6.7	C.2:56
ARF.SE.13	EGL	Aufzeichnung über Quellcode	12207-1:6.24	C.2:56
ARF.SE.14	EGL	Beschreibung der Systemarchitektur und Anforderungsallokation	12207-1:6.25	C.2:53
ARF.SE.15	EGL	Beschreibung der Softwarearchitektur	12207-1:6.12	C.2:54 C.2:55
ARF.SE.16	EGL	Design der Datenhaltung	12207-1:6.4	C.2:101
ARF.SE.17	EGL	Beschreibung des Softwaredesigns	12207-1:6.16	C.2:54 C.2:55
ARF.SE.18	EGL	Aufzeichnung über Modifikationen	12207-1:4.3	C.2:96
ARF.SE.19	EGL	Beschreibung des Schnittstellendesigns	12207-1:6.19	C.2:8
ARF.SE.20	BGL	Test- bzw. Validierungsplan	12207-1:6.27	C.2:59
ARF.SE.21	BGL	Plan des Wartungsablaufs	12207-1:6.8	-
ARF.SE.22	BGL	Plan der Softwareintegration	12207-1:6.18	C.2:65
ARF.SE.23	BGL	Plan für die Ausführung des Produktes	12207-1:6.9	C.2:75
ARF.SE.24	BGL	Plan zur Sicherung der Güte der Software	12207-1:6.20	C.2:25

Weiter auf der nächsten Seite

Referenz	UE-Phase	Artefakt	Quelle ISO/IEC 12207	Bezug ISO/IEC 15504
ARF.SE.25	BGL	Aufzeichnung zur Sicherung der Güte der Software	12207-1:6.21	C.2:28
ARF.SE.26	BGL	Bewertungsaufzeichnungen	12207-1:6.6	C.2:85
ARF.SE.27	BGL	Bericht der Softwareverifikationsergebnisse	12207-1:6.23	C.2:21 C.2:62 C.2:84
ARF.SE.28	BGL	Test- bzw. Validierungsergebnisbericht	12207-1:6.29	C.2:21 C.2:62 C.2:84
ARF.SE.29	BGL	Änderungs-/Modifikationsanfrage	12207-1:6.2	C.2:94
ARF.SE.30	SAF	Plan zur Softwareinstallation	12207-1:4.3	C.2:74
ARF.SE.31	SAF	Ergebnisaufzeichnungen der Softwareinstallation	12207-1:4.3	C.2:106
ARF.SE.32	SAF	Trainingsplan, -materialien und -aufzeichnungen	12207-1:4.3	C.2:88 C.2:89 C.2:90
ARF.SE.33	SAF	Aufzeichnung über Unterstützungsanfragen der Benutzer	12207-1:4.3	C.2:83
ARF.SE.34	SAF	Problemaufzeichnungen	12207-1:6.10	C.2:84
ARF.SE.35	SAF	Aufzeichnung des Softwarekonfigurationsindex	12207-1:6.13	C.2:92
ARF.SE.36	SAF	Managementplan der Softwarekonfiguration	12207-1:6.14	C.2:91
ARF.SE.37	SAF	Aufzeichnung des Softwarekonfigurationsmanagements	12207-1:6.15	C.2:92

Tabelle A.5.: Referenzen der Artefakte des Software Engineering

Referenz	UE-Phase	Artefakt	Quelle Kotler 2006
ARF.MM.1	VBN	Marketinginformationen	-
ARF.MM.2	BGL	Bericht über Zielerreichung	-

Tabelle A.6.: Referenzen der Artefakte des Marketing & der Marktforschung

B. Base Practices

Sammlung der in den Standards ISO/TS 18152 & ISO/IEC 15504 formulierten und in Kapitel 3.1.4 & 3.2 zur intrinsischen Validierung angewandten Base Practices. Die einzelnen Ausprägungen der Verwendung ist ebenfalls in den Tabellen angegeben.

Erläuterungen zu Tabellen B.1 und B.2:

- Referenz: Eindeutige Bezeichnung einer Base Practice im jeweiligen Standard
- Base Practice: Ausformulierte Base Practice
- Verwendung: Ausprägung der Verwendung einer Base Practice: Es handelt sich um eine BP, die bereits als Aktivität im Modell verankert ist (A); es handelt sich um eine BP, die eine organisatorische Rahmenanforderung darstellt (O); es handelt sich um eine BP, aus der sich eine neue Aktivität formulieren lässt (N)
- AKT.UE: Referenz zur jeweiligen Aktivität im Modell

B.1. ISO/TS 18152

Referenz	Base Practice	Verv.	AKT. UE
HS.1	Life cycle involvement		
<i>HS.1.1</i>	<i>Human-system issues in conception</i>		
HS.1.1.BP1	Identify expected context of use of systems	A	14
HS.1.1.BP2	Analyse the system concept	A	14
HS.1.1.BP3	Describe the objectives which the user or user organization wants to achieve through use of the system	A	13
HS.1.1.BP4	Identify and analyse the roles of each group of stakeholders likely to be affected by the system	A	11
HS.1.1.BP5	Perform research into required system usability	A	18
HS.1.1.BP6	Present context and human resources options and constraints to the project stakeholders	O	-
HS.1.1.BP7	Contribute to the business case for the system	N	15
<i>HS.1.2</i>	<i>Human-system issues in development</i>		
HS.1.2.BP1	Generate design options for each aspect of the system related to its use and its effect on stakeholders	A	42
HS.1.2.BP2	Produce user-centred solutions for each design option	A	42
HS.1.2.BP3	Design for customisation	A	42
HS.1.2.BP4	Develop simulation or trial implementation of key aspects of the system for the purpose of testing with users	A	42

Weiter auf der nächsten Seite

Referenz	Base Practice	Verv.	AKT. UE
HS.1.2.BP5	Collect user input on the usability of the developing system	A	18
HS.1.2.BP6	Assess the health and well-being risks to the users of the system	A	22
HS.1.2.BP7	Assess the risks to the community and environment arising from human error in the use of the system	N	16
<i>HS.1.3</i>	<i>Human-system issues in production and utilization</i>		
HS.1.3.BP1	Evolve options and constraints into an implementation strategy covering technical, integration, and planning and manning issues	N	38
HS.1.3.BP2	Identify, specify and produce the infrastructure for the system	N	38
HS.1.3.BP3	Maintain contact with users and the client organization throughout the definition, development and introduction of a system	A	5
HS.1.3.BP4	Build required competencies into training and awareness programmes	A	4
HS.1.3.BP5	Test that the system meets the requirements of the users, the tasks and the environment, as defined in its specification	A	47
HS.1.3.BP6	Analyse feedback on the system during delivery and inform the organization of emerging issues	A	52
<i>HS.1.4</i>	<i>Human-system issues in utilization and support</i>		
HS.1.4.BP1	Produce personnel strategy	O	-
HS.1.4.BP2	Deliver training and other forms of awareness-raising to users and support staff.	O	-
HS.1.4.BP3	Review the system for adherence to applicable human science knowledge, style guides, standards, guidelines, regulations and legislation	A	41
HS.1.4.BP4	Assess the effect of change on the usability of the system	A	43
HS.1.4.BP5	Review the health and well-being risks to the users of the system	N	17
HS.1.4.BP6	Review the risks to the community and environment arising from human error in the use of the system	N	16
HS.1.4.BP7	Take action on issues arising from in-service assessment	O	-
HS.1.4.BP8	Perform research to refine and consolidate operation and support strategy for the system	O	-
<i>HS.1.5</i>	<i>Human-system issues in retirement</i>		
HS.1.5.BP1	Collect and analyse in-service reports to generate updates or lessons learnt for the next version of the system	A	45
HS.1.5.BP2	Identify risks and health and safety issues associated with removal from service and destruction of the system	O	-
HS.1.5.BP3	Define how users will be re-allocated, dismissed, or transferred to other duties	O	-
HS.1.5.BP4	Plan break-up of social structures	O	-
HS.1.5.BP5	Debriefing and retrospective analysis for replacement system	O	-
HS.2	Integrate human factors		
<i>HS.2.1</i>	<i>Human-system issues in business strategy</i>		
HS.2.1.BP1	Define usability as a competitive asset	A	1
HS.2.1.BP2	Set usability, health and safety objectives for systems	O	-
HS.2.1.BP3	Follow competitive situation in the market place	O	-
HS.2.1.BP4	Develop user-centred infrastructure	A	1

Weiter auf der nächsten Seite

Referenz	Base Practice	Verv.	AKT. UE
HS.2.1.BP5	Relate HS issues to business benefits	O	-
<i>HS.2.2</i>	<i>Human-system issues in quality management</i>		
HS.2.2.BP1	Establish and communicate a policy for human-centredness	O	-
HS.2.2.BP2	Include HR and user-centred elements in support and control procedures	O	-
HS.2.2.BP3	Define and maintain HCD and HR infrastructure and resources	O	-
HS.2.2.BP4	Increase and maintain awareness of usability	O	-
HS.2.2.BP5	Develop or provide staff with suitable HS skills	A	4
<i>HS.2.3</i>	<i>Human-system issues in authorisation and control</i>		
HS.2.3.BP.1	Take account of stakeholder and user issues in acquisition activities	A	18
HS.2.3.BP2	Take account of HS issues in financial management	O	-
HS.2.3.BP3	Assess and improve HS capability in processes that affect usability, health and safety	O	-
HS.2.3.BP4	Include HS review and sign-off in all reviews and decisions	O	-
<i>HS.2.4</i>	<i>Management of human-system issues</i>		
HS.2.4.BP1	Develop a plan to achieve and maintain usability throughout the life of the system	O	-
HS.2.4.BP2	Identify the specialist skills required and plan how to provide them	A	4
HS.2.4.BP3	Manage the life cycle plan to address HS issues	A	3
<i>HS.2.5</i>	<i>HF data in trade-off and risk mitigation</i>		
HS.2.5.BP1	Plan and manage use of HF data to mitigate risks related to HS issues	A	22
HS.2.5.BP2	Assess the extent to which usability criteria and other HS requirements are likely to be met by the proposed design	A	57
HS.2.5.BP3	Evaluate the current severity of emerging threats to system usability and other HS risks and the effectiveness of mitigation measures	A	46
HS.2.5.BP4	Take effective mitigation to address risks to system usability	-	-
<i>HS.2.6</i>	<i>User involvement</i>		
HS.2.6.BP1	Identify the HS issues and aspects of the system that require user input	O	-
HS.2.6.BP2	Assess the risks of not involving end users in each evaluation	O	-
HS.2.6.BP3	Define a strategy and plan for user involvement	N	9
HS.2.6.BP4	Select and use the most effective method to elicit user input	A	2
HS.2.6.BP5	Take account of user input and inform users	O	-
<i>HS.2.7</i>	<i>Human-system integration</i>		
HS.2.7.BP1	Develop a common terminology for HS issues with the organization	A	5
HS.2.7.BP2	Facilitate personal and technical interactions related to HS issues	A	33
HS.2.7.BP3	Identify and use the most suitable data formats for exchanging HF data	A	5
HS.2.7.BP4	Customise tools and methods as necessary for particular projects/stages	O	-
HS.2.7.BP5	Identify emerging HS issues	A	14
<i>HS.2.8</i>	<i>Develop and re-use HF data</i>		

Weiter auf der nächsten Seite

Referenz	Base Practice	Verv.	AKT. UE
HS.2.8.BP1	Have a policy for HF data management	O	-
HS.2.8.BP2	Perform research to develop HF data as required	A	22
HS.2.8.BP3	Produce coherent data standards and formats	A	5
HS.2.8.BP4	Define rules for the management of data	A	6
HS.2.8.BP5	Develop and maintain adequate data search methods	O	-
HS.2.8.BP6	Seek and exploit expert guidance and advice on HS issues	A	57
HS.3	Human-centred design		
<i>HS.3.1</i>	<i>Context of use</i>		
HS.3.1.BP1	Define the scope of the context of use for the system	N	10
HS.3.1.BP2	Analyse the tasks and worksystem	A	13
HS.3.1.BP3	Describe the characteristics of the users	A	12
HS.3.1.BP4	Describe the cultural environment/organizational/management regime	A	14
HS.3.1.BP5	Describe the characteristics of any equipment external to the system and the working environment	A	14
HS.3.1.BP6	Describe the location, workplace equipment and ambient conditions	A	14
HS.3.1.BP7	Analyse the implications of the context of use	A	20
HS.3.1.BP8	Present these issues to project stakeholders for use in the development or operation of the system	A	48
<i>HS.3.2</i>	<i>User requirements</i>		
HS.3.2.BP1	Set and agree the expected behaviour and performance of the system with respect to the user	A	29
HS.3.2.BP2	Develop and explicit statement of the user requirements for the system	A	19
HS.3.2.BP3	Analyse the user requirements	A	25
HS.3.2.BP4	Generate and agree on measurable criteria for the system in its intended context of use	A	23
HS.3.2.BP5	Present these requirements to project stakeholders for use in the development or operation of the system	A	48
<i>HS.3.3</i>	<i>Produce design solutions</i>		
HS.3.3.BP1	Distribute functions between the human, machine and organizational elements of the system best able to fulfil each function	A	33
HS.3.3.BP2	Develop a practical model of the users' work from the requirements, context of use, allocation of function and design constraints for the system	A	42
HS.3.3.BP3	Produce designs for the user-related elements of the system that take account of the user requirements, context of use and HF data	A	39
HS.3.3.BP4	Produce a description of how the system will be used	A	44
HS.3.3.BP5	Revise design and safety features using feedback from evaluations	A	43
<i>HS.3.4</i>	<i>Evaluation of use</i>		
HS.3.4.BP1	Plan the evaluation	A	50
HS.3.4.BP2	Identify and analyse the conditions under which a system is to be tested or otherwise evaluated	A	50
HS.3.4.BP3	Check that the system is fit for evaluation	A	50
HS.3.4.BP4	Carry out and analyse the evaluation according to the evaluation plan	A	51, 52, 54, 55, 56, 57, 58, 59

Weiter auf der nächsten Seite

Referenz	Base Practice	Verv.	AKT. UE
HS.3.4.BP5	Understand and act on the results of the evaluation	A	46
HS.4	Human resources		
<i>HS.4.1</i>	<i>Human resources strategy</i>		
HS.4.1.BP1	Decide upon the goals, behaviours and tasks of the organization	A	4
HS.4.1.BP2	Define the global numbers, skills and supporting equipment needed to achieve those tasks	A	4
HS.4.1.BP3	Decide how many people are needed to fulfil the strategy and what ranges of competence they need	A	4
HS.4.1.BP4	Implement an HR strategy that gives the organization a mechanism for implementing and recording lessons learnt	O	-
HS.4.1.BP5	Feed findings back into future HR procurement, training and delivery strategies	O	-
HS.4.1.BP6	Enable and encourage people and teams to work together to deliver the organization's objectives	O	-
<i>HS.4.2</i>	<i>Define standard competencies and identify gaps</i>		
HS.4.2.BP1	Identify current tasking/duty	O	-
HS.4.2.BP2	Analyse gap between existing and future provision	O	-
HS.4.2.BP3	Identify skill requirements for each role		4
HS.4.2.BP4	Predict staff wastage between present and future	O	-
HS.4.2.BP5	Calculate the available staffing, taking account of working hours, attainable effort and non-availability factor	O	-
HS.4.2.BP6	Compare to define gap and communicate requirement to design of staffing solutions	O	-
HS.4.2.BP7	Create capability to meet system requirements in the future (conduct succession planning)	O	-
HS.4.2.BP8	Produce and promulgate a validated statement of shortfall by number and range of competence	O	-
<i>HS.4.3</i>	<i>Design staffing solution and delivery plan</i>		
HS.4.3.BP1	Identify and allocate the functions to be performed	A	7
HS.4.3.BP2	Specify and produce job designs and competence/skills required to be delivered	O	-
HS.4.3.BP3	Calculate the required number of personnel	O	-
HS.4.3.BP4	Generate costed options for delivery of training and/or redeployment	O	-
HS.4.3.BP5	Evolve options and constraints into an optimal implementation plan	O	-
HS.4.3.BP6	Develop and trial training solution with representative users	O	-
HS.4.3.BP7	Deliver final training solutions to designated staff according to agreed timetable	O	-
HS.4.3.BP8	Identify any opportunities for redeployment	O	-
<i>HS.4.4</i>	<i>Evaluate system solutions and obtain feedback</i>		
HS.4.4.BP1	Develop a strategy for data gathering	A	50
HS.4.4.BP2	Provide means for user feedback	O	-
HS.4.4.BP3	Conduct assessments of usability	A	57
HS.4.4.BP4	Interpret the findings	A	52
HS.4.4.BP5	Validate the data	A	48
HS.4.4.BP6	Check that the data are being used	N	59

Tabelle B.1.: Base Practices ISO/TS 18152

B.2. ISO/IEC 15504

Referenz	Base Practice	Verv.	AKT. SE
Primary life cycle processes			
CUS	Customer-supplier process category		
CUS.1	Acquisition process		
CUS.1.BP1	Identify the need	A	6
CUS.1.BP2	Prepare and negotiate contract	O	-
CUS.1.BP3	Monitor the acquisition	O	-
<i>CUS.1.1</i>	<i>Acquisition preparation process</i>		
CUS.1.1.BP1	Identify the need	O	-
CUS.1.1.BP2	Define the requirements	S	-
CUS.1.1.BP3	Prepare acquisition strategy	O	-
CUS.1.1.BP4	Define acceptance criteria	A	21
<i>CUS.1.2</i>	<i>Supplier selection process</i>		
CUS.1.2.BP1	Define acquisition requirements	O	-
CUS.1.2.BP2	Select a supplier	O	-
CUS.1.2.BP3	Prepare and negotiate contract	O	-
<i>CUS.1.3</i>	<i>Supplier monitoring process</i>		
CUS.1.3.BP1	Provide supplier feedback	O	-
CUS.1.3.BP2	Review development with supplier	O	-
CUS.1.3.BP3	Monitor the acquisition	O	-
CUS.1.3.BP4	Monitor supplier	O	-
<i>CUS.1.4</i>	<i>Customer acceptance process</i>		
CUS.1.4.BP1	Evaluate the delivered product	A	66
CUS.1.4.BP2	Accept the delivered product	A	66
CUS.2	Supply process		
CUS.2.BP1	Prepare response	O	-
CUS.2.BP2	Negotiate contract	O	-
CUS.2.BP3	Develop system or software	S	-
CUS.2.BP4	Identify attributes for successful delivery and installation	A	71, 72
CUS.2.BP5	Deliver and install software	A	71, 72
CUS.3	Requirements elicitation process		
CUS.3.BP1	Obtain customer requirements and requests	A	7
CUS.3.BP2	Agree on requirements	N	23
CUS.3.BP3	Establish customer requirements baseline	A	22
CUS.3.BP4	Manage customer requirements changes	N	24
CUS.3.BP5	Understand customer expectations	S	-
CUS.3.BP6	Establish customer query mechanism	A	16
CUS.4	Operation process		
CUS.4.BP1	Software Operation	O	-
CUS.4.BP2	Operation evaluation	A	52, 54
CUS.4.BP3	Support customer	O	-
<i>CUS.4.1</i>	<i>Operational use process</i>		

Weiter auf der nächsten Seite

Referenz	Base Practice	Verv.	AKT. SE
CUS.4.1.BP1	Identify operational risks	O	-
CUS.4.1.BP2	Perform operational testing	A	55
CUS.4.1.BP3	Operate the software	O	-
CUS.4.1.BP4	Review software operation problem	A	53
CUS.4.1.BP5	Resolve operational problems	A	56
CUS.4.1.BP6	Handle user requests	A	48
CUS.4.1.BP7	Document temporary work-arounds	O	-
CUS.4.1.BP8	Monitor system capacity and service	O	-
<i>CUS.4.2</i>	<i>Customer support process</i>		
CUS.4.2.BP1	Provide user training	A	73
CUS.4.2.BP2	Establish product support	O	-
CUS.4.2.BP3	Monitor performance	O	-
CUS.4.2.BP4	Determine customer satisfaction level	O	-
CUS.4.2.BP5	Compare with competitors	O	-
CUS.4.2.BP6	Communicate customer satisfaction	O	-
ENG	Engineering process category		
ENG.1	Development process		
ENG.1.BP1	Define and implement the software or system development process	A	1
ENG.1.BP2	Define and implement the traceability process	A	17
ENG.1.BP3	Define and implement the testing process	A	16
ENG.1.BP4	Define and implement the delivery process	O	-
<i>ENG.1.1</i>	<i>System requirements analysis and design process</i>		
ENG.1.1.BP1	Identify system requirements	A	8, 20
ENG.1.1.BP2	Analyse system requirements	A	8, 9
ENG.1.1.BP3	Describe system architecture	A	30
ENG.1.1.BP4	Allocate requirements	A	26
ENG.1.1.BP5	Develop release strategy	A	2
ENG.1.1.BP6	Communicate system requirements	N	25
ENG.1.1.BP7	Establish traceability	A	17, 59
<i>ENG.1.2</i>	<i>Software requirements analysis process</i>		
ENG.1.2.BP1	Specify software requirements	A	9
ENG.1.2.BP2	Determine operating environment impact	A	11
ENG.1.2.BP3	Evaluate and validate requirements with customer	N	23
ENG.1.2.BP4	Develop validation criteria for software	A	16
ENG.1.2.BP5	Develop release strategy	A	2
ENG.1.2.BP6	Update requirements	N	24
ENG.1.2.BP7	Communicate software requirements	N	25
ENG.1.2.BP8	Evaluate the software requirements	A	18
<i>ENG.1.3</i>	<i>Software design process</i>		
ENG.1.3.BP1	Develop software architectural design	A	26, 39
ENG.1.3.BP2	Design interfaces	A	15, 36, 37
ENG.1.3.BP3	Verify the software design	A	61
ENG.1.3.BP4	Develop detailed design	A	31
ENG.1.3.BP5	Establish traceability	A	17

Weiter auf der nächsten Seite

Referenz	Base Practice	Verv.	AKT. SE
<i>ENG.1.4</i>	<i>Software construction process</i>		
ENG.1.4.BP1	Develop software units	A	32
ENG.1.4.BP2	Develop unit verification procedures	A	33
ENG.1.4.BP3	Verify the software units	A	51
ENG.1.4.BP4	Establish traceability	A	17
<i>ENG.1.5</i>	<i>Software integration process</i>		
ENG.1.5.BP1	Develop software integration strategy	A	63
ENG.1.5.BP2	Develop integrated software item regression test strategy	A	44, 45
ENG.1.5.BP3	Develop tests for integrated software items	A	44, 45
ENG.1.5.BP4	Test integrated software item	A	51
ENG.1.5.BP5	Integrate software item	N	40
ENG.1.5.BP6	Regression test integrated software items	A	51
<i>ENG.1.6</i>	<i>Software testing process</i>		
ENG.1.6.BP1	Develop integrated software test strategy, including regression strategy	A	42, 43
ENG.1.6.BP2	Develop tests for integrated software	A	42, 43
ENG.1.6.BP3	Test integrated software	A	49
ENG.1.6.BP4	Regression test integrated software	A	49
<i>ENG.1.7</i>	<i>System integration and testing process</i>		
ENG.1.7.BP1	Develop system integration and test strategy	N	46
ENG.1.7.BP2	Develop system regression test strategy	N	46
ENG.1.7.BP3	Build aggregates of system units	N	41
ENG.1.7.BP4	Develop tests for system aggregates	N	46
ENG.1.7.BP5	Test system aggregates	A	65
ENG.1.7.BP6	Develop tests for system	N	46
ENG.1.7.BP7	Test integrated system	A	65
ENG.1.7.BP8	Regression test system aggregates or integrated system	A	64
ENG.2	System and software maintenance process		
ENG.2.BP1	Determine maintenance requirements	A	13
ENG.2.BP2	Develop maintenance strategy	N	47
ENG.2.BP3	Analyse user problems and enhancements	A	46
ENG.2.BP4	Determine modifications for next upgrade	A	57
ENG.2.BP5	Implement and test modifications	A	29
ENG.2.BP6	Upgrade user system	A	2
ENG.2.BP7	Retire user system	A	3
Supporting life cycle processes			
SUP	Support process category		
SUP.1	Documentation process		
SUP.1.BP1	Develop documentation policy	O	-
SUP.1.BP2	Establish standards for documents	O	-
SUP.1.BP3	Specify documentation requirements	O	-
SUP.1.BP4	Develop document	O	-
SUP.1.BP5	Check document	O	-
SUP.1.BP6	Distribute document	O	-
SUP.1.BP7	Maintain document	O	-
SUP.2	Configuration management process		

Weiter auf der nächsten Seite

Referenz	Base Practice	Verv.	AKT. SE
SUP.2.BP1	Develop configuration management strategy	O	-
SUP.2.BP2	Establish configuration management system	O	-
SUP.2.BP3	Identify configuration items	O	-
SUP.2.BP4	Maintain configuration item description	O	-
SUP.2.BP5	Manage changes	O	-
SUP.2.BP6	Manage product releases	O	-
SUP.2.BP7	Maintain configuration item history	O	-
SUP.2.BP8	Report configuration status	O	-
SUP.2.BP9	Manage the release and delivery of configuration items	O	-
SUP.3	Quality assurance process		
SUP.3.BP1	Develop quality assurance strategy	O	-
SUP.3.BP2	Establish quality standards	O	-
SUP.3.BP3	Define quality records	O	-
SUP.3.BP4	Assure quality of process activities	A	67, 68
SUP.3.BP5	Assure quality of work products	A	67, 68
SUP.3.BP6	Report quality results	N	70
SUP.3.BP7	Handle deviations	N	70
SUP.4	Verification process		
SUP.4.BP1	Develop verification strategy	O	-
SUP.4.BP2	Conduct verification	A	58
SUP.4.BP3	Determine actions for verification results	N	70
SUP.4.BP4	Track actions for verification results	S	-
SUP.5	Validation process		
SUP.5.BP1	Develop Validation Strategy	O	-
SUP.5.BP2	Perform validation	A	59
SUP.5.BP3	Determine actions for validation results	N	70
SUP.5.BP4	Track actions for validation results	S	-
SUP.6	Joint review process		
SUP.6.BP1	Prepare joint review	A	28
SUP.6.BP2	Establish review criteria	O	-
SUP.6.BP3	Conduct joint management review	S	-
SUP.6.BP4	Conduct joint technical review	S	-
SUP.6.BP5	Conduct joint process review	S	-
SUP.6.BP6	Conduct joint system acceptance review	S	-
SUP.6.BP7	Determine actions for review results	S	-
SUP.6.BP8	Track actions for review results	S	-
SUP.7	Audit Process		
SUP.7.BP1	Develop and implement audit strategy	A	69
SUP.7.BP2	Plan an audit	O	-
SUP.7.BP3	Audit software development activities	S	-
SUP.7.BP4	Audit management activities	S	-
SUP.7.BP5	Audit process performance	S	-
SUP.7.BP6	Audit final products and system	S	-
SUP.7.BP7	Identify corrective actions from the audit report	S	-
SUP.7.BP8	Track actions for audit report	S	-
SUP.8	Problem resolution process		
SUP.8.BP1	Establish problem report system	A	53

Weiter auf der nächsten Seite

Referenz	Base Practice	Verv.	AKT. SE
SUP.8.BP2	Prioritize problems	S	-
SUP.8.BP3	Determine actions for problems	S	-
SUP.8.BP4	Track actions for problems	S	-
SUP.8.BP5	Review and distribute solutions	S	-
SUP.8.BP6	Analyse problem trends	S	-
Organizational	life cycle processes		
MAN	Management process category		
MAN.1	Management process		
MAN.1.BP1	Identify activities and tasks	O	-
MAN.1.BP2	Evaluate feasibility of achieving process	O	-
MAN.1.BP3	Plan and allocate resources and infrastructure	O	-
MAN.1.BP4	Implement activities	O	-
MAN.1.BP5	Monitor performance	O	-
MAN.1.BP6	Review work products and evaluate results	O	-
MAN.1.BP7	Take action on performance deviation	O	-
MAN.1.BP8	Demonstrate successful achievement	O	-
MAN.2	Project management process		
MAN.2.BP1	Define the scope of work	O	-
MAN.2.BP2	Determine development strategy	O	-
MAN.2.BP3	Select software life cycle model	O	-
MAN.2.BP4	Size and estimate tasks and resources	O	-
MAN.2.BP5	Develop work breakdown structure	O	-
MAN.2.BP6	Identify infrastructure requirements	O	-
MAN.2.BP7	Establish project schedule	O	-
MAN.2.BP8	Allocate responsibilities	O	-
MAN.2.BP9	Identify interfaces	O	-
MAN.2.BP10	Establish and implement project plans	O	-
MAN.2.BP11	Track progress against plans	O	-
MAN.2.BP12	Act to correct deviations	O	-
MAN.3	Quality management process		
MAN.3.BP1	Establish quality goals	O	-
MAN.3.BP2	Define overall strategy	O	-
MAN.3.BP3	Identify quality activities	O	-
MAN.3.BP4	Perform quality activities	O	-
MAN.3.BP5	Assess quality	O	-
MAN.3.BP6	Take corrective action	O	-
MAN.4	Risk management process		
MAN.4.BP1	Establish risk management scope	O	-
MAN.4.BP2	Identify risks	O	-
MAN.4.BP3	Analyse and prioritize risks	O	-
MAN.4.BP4	Define risk management strategies	O	-
MAN.4.BP5	Define risk metrics	O	-
MAN.4.BP6	Implement risk management strategies	O	-
MAN.4.BP7	Assess results of risk management strategies	O	-
MAN.4.BP8	Take corrective action	O	-
ORG	Organization process category		
ORG.1	Organizational alignment process		
ORG.1.BP1	Develop a strategic vision	O	-
ORG.1.BP2	Deploy vision	O	-

Weiter auf der nächsten Seite

Referenz	Base Practice	Verv.	AKT. SE
ORG.1.BP3	Develop a quality culture	O	-
ORG.1.BP4	Build and empower teams	O	-
ORG.1.BP5	Provide incentives	O	-
ORG.2	Improvement process		
ORG.2.BP1	Define organizational processes	O	-
ORG.2.BP2	Deploy the processes	O	-
ORG.2.BP3	Assess the deployed processes	O	-
ORG.2.BP4	Improve the standard processes	O	-
<i>ORG.2.1</i>	<i>Process establishment process</i>		
ORG.2.1.BP1	Define goals	O	-
ORG.2.1.BP2	Identify activities, roles, authorities & responsibilities	O	-
ORG.2.1.BP3	Define and document the processes performed in the organization	O	-
ORG.2.1.BP4	Establish organisational policies	O	-
ORG.2.1.BP5	Establish performance expectations	O	-
ORG.2.1.BP6	Deploy the process	O	-
ORG.2.1.BP7	Check the standard processes deployment	O	-
ORG.2.1.BP8	Capture process data	O	-
ORG.2.1.BP9	Maintain the standard processes	O	-
<i>ORG.2.2</i>	<i>Process assessment process</i>		
ORG.2.2.BP1	Determine the assessment method	O	-
ORG.2.2.BP2	Define assessment goals	O	-
ORG.2.2.BP3	Define the assessment plan	O	-
ORG.2.2.BP4	Plan the assessment	O	-
ORG.2.2.BP5	Perform the assessment to collect data	O	-
ORG.2.2.BP6	Validate the data	O	-
ORG.2.2.BP7	Identify strengths and weaknesses	O	-
ORG.2.2.BP8	Maintain the assessment results	O	-
ORG.2.2.BP9	Exploit the assessment result	O	-
ORG.2.2.BP10	Report the assessment result	O	-
<i>ORG.2.3</i>	<i>Process improvement process</i>		
ORG.2.3.BP1	Identify improvement opportunities	O	-
ORG.2.3.BP2	Define scope of improvement activities	O	-
ORG.2.3.BP3	Understand the process	O	-
ORG.2.3.BP4	Identify improvements	O	-
ORG.2.3.BP5	Prioritize improvements	O	-
ORG.2.3.BP6	Define measures of impact	O	-
ORG.2.3.BP7	Change the process	O	-
ORG.2.3.BP8	Confirm the improvement	O	-
ORG.2.3.BP9	Deploy improvement	O	-
ORG.3	Human resource management process		
ORG.3.BP1	Identify human resource needs	O	-
ORG.3.BP2	Develop or acquire training	O	-
ORG.3.BP3	Train personnel	O	-
ORG.3.BP4	Recruit qualified staff	O	-
ORG.3.BP5	Define evaluation criteria	O	-
ORG.3.BP6	Evaluate staff performance	O	-
ORG.3.BP7	Provide feedback on performance	O	-
ORG.3.BP8	Maintain staff records	O	-

Weiter auf der nächsten Seite

Referenz	Base Practice	Verv.	AKT. SE
ORG.3.BP9	Define project teams	O	-
ORG.3.BP10	Empower project teams	O	-
ORG.3.BP11	Maintain project team interactions	O	-
ORG.4	Infrastructure process		
ORG.4.BP1	Identify software engineering environment requirements	O	-
ORG.4.BP2	Provide a software engineering environment	O	-
ORG.4.BP3	Provide support for individuals using the software engineering infrastructure	O	-
ORG.4.BP4	Maintain software engineering environment	O	-
ORG.4.BP5	Provide a workspace conducive to productive performance	O	-
ORG.4.BP6	Ensure data integrity and security	O	-
ORG.4.BP7	Provide remote access facility	O	-
ORG.5	Measurement process		
ORG.5.BP1	Establish metrics for process management	O	-
ORG.5.BP2	Establish metrics for the quality of work products	O	-
ORG.5.BP3	Conduct quantitative process management	O	-
ORG.5.BP4	Measure the quality of work products	O	-
ORG.5.BP5	Make measurement data available for decision-making	O	-
ORG.5.BP6	Define benchmark	O	-
ORG.5.BP7	Benchmark processess	O	-
ORG.6	Reuse process		
ORG.6.BP1	Define organizational reuse strategy	O	-
ORG.6.BP2	Establish reuse library	O	-
ORG.6.BP3	Identify reusable entities	O	-
ORG.6.BP4	Develop reusable entities	O	-
ORG.6.BP5	Keep reusable entities stable and consistent	O	-
ORG.6.BP6	Report and certify reusable entities and domain knowledge	O	-
ORG.6.BP7	Inform potential users about reusable entities and domain knowledge	O	-

Tabelle B.2.: Base Practices ISO/IEC 15504

C. Artefakt Lebenszyklus

Abbildung des Lebenszyklus von Artefakten als Ausgangsbasis (*Input*) beziehungsweise Ergebnis (*Output*) einer Aktivität.

Erläuterungen zu Tabellen C.1 und C.2:

- Zeilen: Aktivitäten mit Referenz zu Anhang A.1
- Spalten: Artefakte mit Referenz zu Anhang A.2
- I = Input: Artefakt als Ausgangsbasis für eine Aktivität
- O = Output: Artefakt als Ergebnis einer Aktivität

	ARF.UE.1	ARF.UE.2	ARF.UE.3	ARF.UE.4	ARF.UE.5	ARF.UE.6	ARF.UE.7	ARF.UE.8	ARF.UE.9	ARF.UE.10	ARF.UE.11	ARF.UE.12	ARF.UE.13	ARF.UE.14	ARF.UE.15	ARF.UE.16	ARF.UE.17	ARF.UE.18	ARF.UE.19	ARF.UE.20	ARF.UE.21	ARF.UE.22	ARF.UE.23	ARF.UE.24	ARF.UE.25	ARF.UE.26	ARF.UE.27	ARF.UE.28	ARF.UE.29	ARF.UE.30	ARF.UE.31
AKT.UE.1	O		O																												
AKT.UE.2	O																														
AKT.UE.3	O																														
AKT.UE.4	O																														
AKT.UE.5	O	O																													
AKT.UE.6	O																														
AKT.UE.7	O																														
AKT.UE.8	O																														
AKT.UE.9	O																														
AKT.UE.10				O	O																										
AKT.UE.11				O	O																										
AKT.UE.12				O	O																										
AKT.UE.13				O	O																										
AKT.UE.14				O	O																										
AKT.UE.15				O	O																										
AKT.UE.16				O	O																										
AKT.UE.17				O	O																										
AKT.UE.18				I	O																										
AKT.UE.19				I					O																						
AKT.UE.20				I					O																						
AKT.UE.21				I	I				O																						
AKT.UE.22				I					O	I	I	I	I																		
AKT.UE.23				I		O	O	O	O																						
AKT.UE.24				I					O																						
AKT.UE.25									IO																						
AKT.UE.26									IO																						
AKT.UE.27									IO																						
AKT.UE.28						I		I						O																	
AKT.UE.29				I		I		I						O	I																
AKT.UE.30														O		I															
AKT.UE.31														O																	
AKT.UE.32														O																	
AKT.UE.33														O																	
AKT.UE.34														O																	
AKT.UE.35														O			I														
AKT.UE.36														O																	
AKT.UE.37														O																	
AKT.UE.38									I																						
AKT.UE.39				I		I		I						I	I						O										
AKT.UE.40														I							O										
AKT.UE.41														I	I			I	I	I	O										
AKT.UE.42																					I	O									
AKT.UE.43																					O										
AKT.UE.44																					I	O					I	I			
AKT.UE.45					O																										
AKT.UE.46						I	I																								
AKT.UE.47						I	I	I																							
AKT.UE.48																															
AKT.UE.49																															
AKT.UE.50																															
AKT.UE.51																															
AKT.UE.52																															
AKT.UE.53																															
AKT.UE.54																															
AKT.UE.55																															
AKT.UE.56									I																						
AKT.UE.57																															
AKT.UE.58									I																						
AKT.UE.59																															

Tabelle C.1.: Artefakt-Lebenszyklus in den Aktivitäten des Usability Engineering

D. Gemeinsamkeiten

Sammlung, der im Modell in Kapitel 3.1.2 ermittelten Gemeinsamkeiten beziehungsweise Ähnlichkeiten zwischen Aktivitäten und Artefakten des Usability Engineering und Software Engineering.

Erläuterungen zu Tabellen D.1 und D.2:

- Nr: Eindeutige Referenznummer
- Disziplin: Disziplin, aus der die jeweilige Aktivität bzw. das jeweilige Artefakt stammt
- Aktivität/BPs: Referenz zur jeweiligen Aktivität bzw. zur jeweiligen Base Practice
- Artefakt/WPs: Referenz zum jeweiligen Artefakt bzw. zum jeweiligen Work Product der Prozessbewertungsstandards
- Aussage: Formulierung der Aktivität bzw. des Artefakts im jeweiligen Standard
- Ausprägung: Aussage über die Ausprägung der Gemeinsamkeit: Sind die Aktivitäten bzw. Artefakte *identisch*; sind sie von ihrem Inhalt her *ähnlich*; *ergänzen* sie sich gegenseitig; sind sie vom Prinzip her ähnlich, fokussieren jedoch jeweils einen anderen *Schwerpunkt*; ist die Aktivität bzw. das Artefakt jeweils in einer Aktivität bzw. einem Artefakt der anderen Disziplin enthalten (*SE in UE*, *UE in SE*)

D.1. Aktivitäten

Nr.	Disz.	Aktivität / BPs	Aussage	Auspräg.
G.1	UE	AKT.UE.1	„[...] the decision has been made to use human-centred development, [...]“	ergänzen
		HS.2.1.BP1	„Define usability as a competitive asset“	
		HS.2.1.BP4	„Develop user-centred infrastructure“	
	SE	AKT.SE.1	„[...] define or select a software life cycle model appropriate to the scope, magnitude and complexity of the project.“	

Weiter auf der nächsten Seite

Nr.	Disz.	Aktivität / BPs	Aussage	Auspräg.
		ENG.1.BP1	„Define and implement the software or system development process“	
	MM	AKT.MM.1	„Markt- und kundenorientiertes Denken“	
G.2	UE	AKT.UE.11	„Relevant groups shall be identified and their relationship with the proposed development described in terms of key goals and constraints.“	ähnlich
		HS.1.1.BP4	„Identify and analyse the roles of each group of stakeholders likely to be affected by the system“	
	SE	AKT.SE.4	„User documentation“	
		AKT.SE.5	„[...] update user documentation as necessary.“	
	MM	AKT.MM.2	„Käufer suchen“	
G.3	UE	AKT.UE.12	„[...] relevant characteristics of the users shall be identified.“	ergänzen
		HS.3.1.BP3	„Describe the characteristics of the users“	
	MM	AKT.MM.5	„Ein Kauf wird stark durch kulturelle, soziale, persönliche und psychologische Charakteristika beeinflusst. [...] über deren Wirkungen im Klaren sein.“	
G.4	UE	AKT.UE.14	„[...] the technical environment, including the hardware, software and materials, shall be identified.“	ergänzen
		HS.1.1.BP1	„Identify expected context of use of system“	
		HS.1.1.BP2	„Analyse the system concept“	
		HS.2.7.BP5	„Identify emerging HS issues“	
		HS.3.1.BP4	„Describe the cultural environment/organizational/management regime“	
		HS.3.1.BP5	„Describe the characteristics of any equipment external to the system and the working environment“	
		HS.3.1.BP6	„Describe the location, workplace equipment and ambient conditions“	
	MM	AKT.MM.4	„Marktsegmentierung“	
		AKT.MM.6	„Marko-Umfeld: Demografische Entwicklung, Ökonomie, Umwelt, Technologie, Politik, Kultur“	
		AKT.MM.7	„Märkte: Endverbrauchermärkte, Industriegütermärkte, Handelsmärkte, Märkte öffentlicher Institutionen, Staatliche Nachfragemärkte, Internationale Märkte“	
		AKT.MM.8	„Konkurrenzanalyse“	
		AKT.MM.9	„Öffentlichkeit: Finanzsektor, Medien, Staat, Bürgerinitiativen, Lokale Interessengruppen, Allgemeine Öffentlichkeit, Unternehmensinterne Öffentlichkeit“	

Weiter auf der nächsten Seite

Nr.	Disz.	Aktivität / BPs	Aussage	Auspräg.
		AKT.MM.10	„Organisationsbezogene Einflussgrößen“	
G.5	UE	AKT.UE.18	„User and other stakeholder needs should be identified, [...]“	ergänzen
		HS.1.1.BP5	„Perform research into required system usability“	
		HS.1.2.BP5	„Collect user input on the usability of the developing system“	
		HS.2.3.BP1	„Take account of stakeholder and user issues in acquisition activities“	
	MM	AKT.MM.3	„[...] deren Bedürfnisse erkennen“	
G.6	UE	AKT.UE.19	„Deriving user requirements“	ergänzen
		HS.3.2.BP2	„Develop and explicit statement of the user requirements for the system“	
	SE	AKT.SE.19	„[...] user requirements [...]“	
		AKT.SE.10	„User operation and execution requirements“	
		AKT.SE.11	„[...] operations, and maintenance requirements [...]“	
		ENG.1.2.BP2	„Determine operating environment impact“	
		AKT.SE.14	„User maintenance requirements.“	
		AKT.SE.13	„[...] operations, and maintenance requirements [...]“	
		ENG.2.BP1	„Determine maintenance requirements“	
G.7	UE	AKT.UE.20	„[...] requirements derived from user needs and the context of use [...]“	ergänzen
		HS.3.1.BP7	„Analyse the implications of the context of use“	
	SE	AKT.SE.7	„[...] business [...] requirements [...]“	
		CUS.3.BP1	„Obtain customer requirements and requests“	
G.8	UE	AKT.UE.22	„[...] requirements arising from relevant ergonomics and user interface knowledge, standards and guidelines [...]“	ähnlich
		HS.1.2.BP6	„Assess the health and well-being risks to the users of the system“	
		HS.2.5.BP1	„Plan and manage use of HF data to mitigate risks related to HS issues“	
		HS.2.8.BP2	„Perform research to develop HF data as required“	
	SE	AKT.SE.9	„[...] safety, security, human-factors engineering (ergonomics) [...]“, „Safety specifications [...]“, „Security specifications [...]“, „Human-factors engineering (ergonomics) [...]“	
G.9	UE	AKT.UE.24	„[...] requirements derived from organizational requirements that directly affect the user [...]“	ähnlich

Weiter auf der nächsten Seite

Nr.	Disz.	Aktivität / BPs	Aussage	Auspräg.
	SE	AKT.SE.12	„[...] organizational [...] requirements [...]“	
		ENG.1.1.BP2	„Analyse the system requirements“	
		ENG.1.2.BP1	„Specify software requirements“	
G.10	UE	AKT.UE.26	„The rationales, factors and weighting of human system issues for use in trade-offs should be documented so that they can be understood in the future.“	identisch
	SE	AKT.SE.17	„Traceability [...]“	
		ENG.1.BP2	„Define and implement the traceability process“	
		ENG.1.1.BP7	„Establish traceability“	
		ENG.1.3.BP5	„Establish traceability“	
		ENG.1.4.BP4	„Establish traceability“	
G.11	UE	AKT.UE.27	„Ensuring the quality of user requirements specifications [...] stated in terms that permit subsequent testing [...] verified by the relevant stakeholders [...] internally consistent [...] updated as necessary during the life of the project.“	ähnlich
	SE	AKT.SE.23	-	
		CUS.3.BP2	„Agree on requirements“	
		ENG.1.2.BP3	„Evaluate and validate requirements with customer“	
		AKT.SE.22	„Qualification requirements“	
		CUS.3.BP3	„Establish customer requirements baseline“	
		AKT.SE.16	„Testability“	
		CUS.3.BP6	„Establish customer query mechanism“	
		ENG.1.BP3	„Define and implement the testing process“	
		ENG.1.2.BP4	„Develop validation criteria for software“	
		AKT.SE.18	„Consistency“	
		ENG.1.2.BP8	„Evaluate the software requirements“	
G.12	UE	AKT.UE.38	-	ähnlich
		HS.1.3.BP1	„Evolve options and constraints into an implementation strategy covering technical, integration, and planning and manning issues“	
		HS.1.3.BP2	„Identify, specify and produce the infrastructure for the system“	
	SE	AKT.SE.30	„A top-level architecture of the system shall be established.“	
		ENG.1.1.BP3	„Describe system architecture“	
G.13	UE	AKT.UE.41	„[...] there is a substantial body of ergonomics and user interface knowledge, standards and guidelines which should be used to inform the design of both hardware and software.“	ergänzen

Weiter auf der nächsten Seite

Nr.	Disz.	Aktivität / BPs	Aussage	Auspräg.
		HS.1.4.BP3	„Review the system for adherence to applicable human science knowledge, style guides, standards, guidelines, regulations and legislation“	
	MM	AKT.MM.12	„Bereitstellen von Marketinginformationen“	
G.14	UE	AKT.UE.46	„[...] provide feedback on strengths and weaknesses of the design solution from the user's perspective [...]“	ergänzen
		HS.2.5.BP3	„Evaluate the current severity of emerging threats to system usability and other HS risks and the effectiveness of mitigation measures“	
		HS.3.4.BP5	„Understand and act on the results of the evaluation“	
	SE	AKT.SE.48	„[...] provide assistance and consultation to the users as requested.“, „[...] forward user requests, as necessary, to the Maintenance Process for resolution.“	
		CUS.4.1.BP1	„Identify operational risks“	
G.15	UE	AKT.UE.54	„User-based testing [...]“	Schwerp.
		AKT.UE.55	„[...] assess whether usability objectives, including measureable usability performance and satisfaction criteria, have been met in the intended context or context of use.“	
		AKT.UE.56	„[...] field validation [...]“	
		HS.3.4.BP4	„Carry out and analyse the evaluation according to the evaluation plan“	
	MM	AKT.MM.14	„Feststellen der Zielerreichung“	
	SE	AKT.SE.52	„[...] develop a plan and set operational standards [...]“	
		CUS.4.BP2	„Operation evaluation“	
		AKT.SE.53	„[...] establish procedures for receiving, recording, resolving, tracking problems, and providing feedback.“	
		CUS.4.1.BP4	„Review software operation problem“	
		SUP.8.BP1	„Establish problem report system“	
		AKT.SE.54	„[...] establish procedures for testing the software product in its operation environment [...]“	
		CUS.4.BP2	„Operation evaluation“	
		AKT.SE.55	„[...] perform operational testing [...]“	
		CUS.4.1.BP2	„Perform operational testing“	
		AKT.SE.66	„[...] support the acquirer's acceptance review and testing of the software product. Acceptance review and testing [...]“	
		CUS.1.4.BP1	„Evaluate the delivered product“	
		CUS.1.4.BP2	„Accept the delivered product“	

Weiter auf der nächsten Seite

Nr.	Disz.	Aktivität / BPs	Aussage	Auspräg.
G.16	UE	AKT.UE.57	„Inspection-based evaluation“	Schwerp.
		HS.2.5.BP2	„Assess the extent to which usability criteria and other HS requirements are likely to be met by the proposed design“	
		HS.2.8.BP6	„Seek and exploit expert guidance and advice on HS issues“	
		HS.3.4.BP4	„Carry out and analyse the evaluation according to the evaluation plan“	
		HS.4.4.BP3	„Conduct assessments of usability“	
	SE	AKT.SE.58	„Appropriateness of design standards and methods used“	
		SUP.4.BP2	„Conduct verification“	
		AKT.SE.59	„Traceability [...]“, „Consistency [...]“	
		ENG.1.1.BP7	„Establish traceability“	
		SUP.4.BP2	„Conduct verification“	
		AKT.SE.61	„Feasibility of the software items fulfilling their allocated requirements“	
		ENG.1.3.BP3	„Verify the software design“	
G.17	UE	AKT.UE.59	-	ähnlich
		HS.4.4.BP6	„Check that the data are being used“	
	SE	AKT.SE.70	-	
		SUP.3.BP6	„Report quality results“	
		SUP.3.BP7	„Handle deviations“	
		SUP.4.BP3	„Determine actions for verification results“	
		SUP.5.BP3	„Determine actions for validation results“	

Tabelle D.1.: Gemeinsamkeiten zwischen Aktivitäten und deren Ausprägung

D.2. Artefakte

Nr.	Disz.	Artefakt / WPs	Aussage	Auspräg.
G.18	UE	ARF.UE.4	„Context of use description“	SE in UE
		A.26:110	„Context of use statement“	
	SE	ARF.SE.8	„User documentation description“	
		C.2:105	„Customer documentation“	
G.19	UE	ARF.UE.9	„User requirements specification“	ergänzen
		A.26:52	„Requirement specification“	
	SE	ARF.SE.9	„Software requirements description“	
		C.2:52	„Requirement specification“	
		ARF.SE.10	„System requirements specification“	
		C.2:53	„System design / architecture“	
G.20	UE	ARF.UE.20	„[...] other aspects of the context of use, [...]“	ähnlich
		A.26:46	„Market analysis record/report“	
	MM	ARF.MM.1	„Marketinginformationen“	

Weiter auf der nächsten Seite

Nr.	Disz.	Artefakte WPs	Aussage	Auspräg.
G.21	UE	ARF.UE.27	„[...] feedback on strengths and weaknesses of the design solution [...]“	ähnlich
		A.26:31	„Review record“	
	SE	ARF.SE.29	„Change request or modification request“	
		C.2:94	„Change request“	
G.22	UE	ARF.UE.30	„[...] requirements have been achieved [...]“	ähnlich
	MM	ARF.MM.2	„Bericht über Zielerreichung“	

Tabelle D.2.: Gemeinsamkeiten zwischen Artefakten und deren Ausprägung

E. Konformitäts- und Rahmenanforderungen

Sammlung, der in Kapitel 3.3.2 beschriebenen Konformitäts- und Rahmenanforderungen nach Nebe (2009), sowie die Ergebnisse des Abgleiches mit dem Integrationsmodell dieser Arbeit.

Erläuterungen zu Tabelle E.1:

- Nr: Eindeutige Referenznummer
- UE-Phase: Einordnung in die jeweilige Phase des UE
- Anforderung: Ausformulierte Konformitäts- bzw. Rahmenanforderung
- Referenz: Referenz zum Ursprung der Anforderung in Nebe (2009)
- Relevanz: Anzahl der Erwähnungen einer Anforderung durch Experten nach Nebe (2009)
- Abgleich: Übereinstimmung einer Anforderung mit einer Aktivität im Integrationsmodell dieser Arbeit. Existiert keine Übereinstimmung, ist der Anforderung eine in Kapitel 3.3.3 definierten Kategorie zugeordnet (Anforderung bezieht sich auf den *Prozess* als Ganzes und muss projektspezifisch sichergestellt werden; Anforderung bezieht sich auf den Prozess und ist durch das Modell *erfüllt*; aus der Anforderung wurde eine *neue* Aktivität formuliert und im Modell integriert)

NR	UE-Phase	Anforderung	Ref.	Rel.	Abgleich
KoRa.A1	1 CoU	Die Kontextanalyse bezieht sich auf den originären Kontext der Nutzer (Aufgaben, Ziele, Charakteristiken der Nutzeraufgaben und der Umgebung, etc.), unabhängig einer vorhandenen Umsetzung durch ein System.	I1	4	AKT.UE.11 AKT.UE.12 AKT.UE.13 AKT.UE.14
KoRa.A2	1 CoU	Es existiert ein Verständnis für die Notwendigkeit der Analyse bei allen Prozessbeteiligten, insbesondere dem Management.	I10	4	AKT.UE.1

Weiter auf der nächsten Seite

NR	UE-Phase	Anforderung	Ref.	Rel.	Abgleich
KoRa.A3	1 CoU	Die Kontextanalyse ist Bestandteil des Prozessmodells.	I11	3	erfüllt
KoRa.A4	1 CoU	Die Kontextanalyse wird zeitlich früh, vor Beginn der Konzeption, durchgeführt.	I12	2	erfüllt
KoRa.A5	1 CoU	Die Bereitstellung ausreichender Kapazitäten und Zeit zur Durchführung der Analyse ist sichergestellt.	I13	4	AKT.UE.7 AKT.UE.8
KoRa.A6	1 CoU	Die Kontextinformation ist hinreichende Grundlage für den Designprozess (PDS).	I14	2	AKT.UE.20 AKT.UE.59
KoRa.A7	1 CoU	Der Anspruch der Kontextanalyse ist, so lange zu iterieren, bis nach Möglichkeit alle Unvollständigkeiten und Widersprüche aufgehoben sind.	I15	1	Prozess
KoRa.A8	1 CoU	Die Analysten verfügen über Erfahrungen in der Anwendung der Methodik zur Analyse des Nutzungskontextes (etw. „können“ anstatt nur „kennen“; nicht nur theoretisch erlernt sondern praktisch erprobt).	I16	6	AKT.UE.4
KoRa.A9	1 CoU	Die Kontextanalyse wird im richtigen, originären Kontext durchgeführt.	I17	3	neu
KoRa.A10	1 CoU	Die Kontextanalyse wird mit repräsentativen und geeigneten Nutzern (und mit Repräsentanten aller Nutzergruppen) durchgeführt.	I18	3	AKT.UE.9
KoRa.A11	1 CoU	Die Evaluation der Kontextinformation erfolgt mit repräsentativen Nutzern und Kunden.	I19	3	neu
KoRa.A12	1 CoU	Zusammenhänge von Nutzern, Aufgaben, Ziele, Charakteristiken der Nutzeraufgaben und deren Umgebung, sowie den zugehörigen Rahmenbedingungen werden abgebildet (Umgebung, sowohl physisch als auch organisatorisch).	I2	4	neu
KoRa.A13	1 CoU	Die Kontextinformation ist (für Dritte) verständlich und nachvollziehbar formuliert.	I3	5	neu
KoRa.A14	1 CoU	Die Kontextinformation ist hinreichende Grundlage für das Ableiten von Nutzungsanforderungen.	I4	2	AKT.UE.20
KoRa.A15	1 CoU	Die Kontextinformation sind die Arbeitsgrundlage für jegliche weitere Aktivitäten im Prozess.	I42	1	AKT.UE.59
KoRa.A16	1 CoU	Die Kontextinformation dienen als Validierungsgrundlage für die Nutzeranforderungen (UR & EoU).	I43	2	AKT.UE.27
KoRa.A17	1 CoU	Die Kontextinformationen dienen als Validierungsgrundlage für die Lösung (EoU).	I44	1	neu

Weiter auf der nächsten Seite

NR	UE-Phase	Anforderung	Ref.	Rel.	Abgleich
KoRa.A18	1 CoU	Die Kontextinformation bildet Fakten ab und stellt keine Interpretation der Situation dar.	I5	1	neu
KoRa.A19	1 CoU	Die Kontextanalyse ist eine fortlaufende Aktivität entlang des gesamten Entwicklungsprozesses, bei der bereits Teilergebnisse zur Weiterverarbeitung verwendet werden.	I52	3	AKT.UE.8
KoRa.A20	1 CoU	Das Wissen über die Kontextinformation wird hinreichend kommuniziert.	I56	2	AKT.UE.5
KoRa.A21	1 CoU	Die Kontextanalyse und die resultierenden Informationen (Kontextinformation) werden in Bezug zu den organisatorischen bzw. Kundenzielen gesetzt.	I57	2	neu
KoRa.A22	1 CoU	Die Ergebnisse der Kontextanalyse sind der Input für den nächsten Prozessschritt und als solcher im Modell verankert.	I6	2	erfüllt
KoRa.A23	1 CoU	Durch Iteration der Analyse wird die Widerspruchsfreiheit und Vollständigkeit (bzgl. des Kontextes & der Aufgabe) sichergestellt. D.h. es werden keine (bzw. nur geringfügige) Veränderungen im Kontext mehr identifiziert (Rahmenbedingungen, Arbeitsablauf, etc.). Erfolgskriterium für die Vollständigkeit ist das Erkennen schlüssiger Muster (z.B. im Arbeitsablauf der Nutzer bei der Erledigung einer Aufgabe).	I7	2	AKT.UE.8
KoRa.A24	1 CoU	Die Kontextinformation repräsentiert die Sichtweise der Nutzer und keine technische Sichtweise.	I8	1	neu
KoRa.A25	1 CoU	Die Eigenschaften der zukünftigen Nutzer, ihrer Aufgaben sowie ihrer Interaktionen mit anderen Nutzern oder Systemen werden dokumentiert (bzw. hinreichend kommuniziert).	K1	5	AKT.UE.5 AKT.UE.6
KoRa.A26	1 CoU	Die reale Nutzungsumgebung des Systems inklusive der Faktoren welche die Arbeit der Nutzer mit dem System beeinflussen werden beschrieben (formuliert).	K2	5	AKT.UE.14
KoRa.A27	1 CoU	Die Auswirkungen des Nutzungskontextes auf das System werden in den Rahmenbedingungen und Anforderungen für das System berücksichtigt.	K3	5	AKT.UE.20
KoRa.A28	1 CoU	Nicht alles muss dokumentiert werden - vielmehr muss die Kommunikation dieses Wissens sichergestellt/garantiert sein.	K4	2	AKT.UE.5
KoRa.A29	1 CoU	Die Kontextinformationen werden im Hinblick auf Anforderungen analysiert.	K7	1	AKT.UE.20

Weiter auf der nächsten Seite

NR	UE-Phase	Anforderung	Ref.	Rel.	Abgleich
KoRa.A30	1 CoU	Der Rahmen für den Nutzungskontext wird abgesteckt (z. B. Zielgruppe, Aufgaben, ...)	K8	5	AKT.UE.10
KoRa.A31	1 CoU	Die Aufgaben und das Arbeitssystem werden analysiert.	K9	5	AKT.UE.13
KoRa.A32	1 CoU	Die Eigenschaften von Nutzern werden beschrieben.	K10	5	AKT.UE.12
KoRa.A33	1 CoU	Die kulturelle Umgebung, die Organisationsstruktur sowie Management-Modelle werden beschrieben.	K11	5	AKT.UE.14
KoRa.A34	1 CoU	Eigenschaften von Geräten oder Systemen, die mit dem zukünftigen System gemeinsam genutzt werden sollen, werden beschrieben.	K12	5	AKT.UE.14
KoRa.A35	1 CoU	Der Ort, die vorhandene Arbeitsplatzausstattung sowie Umgebungsfaktoren werden beschrieben.	K13	5	AKT.UE.14
KoRa.A36	1 CoU	Die Auswirkungen des Nutzungskontexts werden untersucht/abgeschätzt.	K14	5	AKT.UE.16
KoRa.A37	1 CoU	Die Ergebnisse werden Projekt-Stakeholdern demonstriert / zur Verfügung gestellt, um sie in der Entwicklung und dem Betrieb des Systems zu nutzen.	K15	5	neu
KoRa.A38	1 CoU	Nicht alle Kontextinformationen müssen dokumentiert werden aber die Kommunikation des Wissens an alle relevanten Stakeholder muss sichergestellt sein. Die Wissensträger müssen stets greifbar/verfügbar sein.	K16	2	AKT.UE.5
KoRa.A39	1 CoU	Die erhobenen und analysierten Kontextinformationen werden evaluiert.	K18	1	neu
KoRa.A40	1 CoU	Die Ergebnisse (Dokumente) werden mit Eigenschaften versehen, die zu erfüllen sind, d.h. die Inhalte werden klar definiert und an messbare Qualitätsattribute gekoppelt (z.B: Umfang, Detailgrad, Richtlinien, etc.).	K19	1	AKT.UE.6
KoRa.A41	1 CoU	Die analysierten Aufgaben und Arbeitssysteme werden hinreichend beschrieben (kommuniziert).	K21	1	AKT.UE.5
KoRa.A42	2 UR	Nutzeranforderungen haben eine bestimmte Formulierungsqualität; sie sind nachvollziehbar und verständlich für alle am Prozess beteiligten.	I20	2	AKT.UE.27
KoRa.A43	2 UR	Die Nutzeranforderungen sind systemneutral (unabhängig einer Lösung) formuliert.	I21	2	neu
KoRa.A44	2 UR	Alle Nutzeranforderungen basieren auf einem Erfordernis (und sind somit valide).	I22	3	AKT.UE.18

Weiter auf der nächsten Seite

NR	UE-Phase	Anforderung	Ref.	Rel.	Abgleich
KoRa.A45	2 UR	Die Nutzeranforderungen sind ausreichend präzise auf Ebene der Aufgabe formuliert.	I23	5	AKT.UE.27
KoRa.A46	2 UR	Die Nutzeranforderungen sind als solche nicht interpretierbar.	I24	1	Prozess
KoRa.A47	2 UR	Das System wird gegen Nutzeranforderungen (bzw. der daraus abgeleiteten Systemspezifikation) validiert (UR & EoU).	I25	1	AKT.UE.47
KoRa.A48	2 UR	Die Erhebung der Nutzeranforderungen erfolgt iterativ (bis keine neuen Erfordernisse mehr identifiziert werden).	I26	3	AKT.UE.8
KoRa.A49	2 UR	Es existieren ausreichend Kapazitäten und Zeit zur Erhebung und Evaluation der Nutzeranforderungen.	I27	2	AKT.UE.8
KoRa.A50	2 UR	Nutzeranforderungen werden mit repräsentativen Nutzern erhoben UND mit repräsentativen Nutzern, sowie mit dem Kunden, evaluiert.	I28	5	AKT.UE.9 AKT.UE.21
KoRa.A51	2 UR	Die Nutzeranforderungen sind die Basis für den nächsten Prozessschritt (PDS) und als solcher im Prozess verankert.	I29	3	erfüllt
KoRa.A52	2 UR	Die Nutzeranforderungen und das Verständnis über die Kontextinformation werden verständlich und nachvollziehbar an alle im Prozess beteiligten Personen, inkl. dem Management, kommuniziert und vermittelt.	I30	2	AKT.UE.5
KoRa.A53	2 UR	Die Analysten verfügen über Erfahrungen in der Anwendung der Methodik (etw. „können“ anstatt nur „kennen“; nicht nur theoretisch erlernt sondern praktisch erprobt) und können Nutzeranforderungen auf Grundlage der Kontextinformation ableiten und abwägen.	I31	3	AKT.UE.4
KoRa.A54	2 UR	Die Nutzeranforderungen sind Grundlage für eine valide Systemspezifikation und die Modellierung (aus Sicht der Nutzung) (PDS).	I45	2	erfüllt
KoRa.A55	2 UR	Eine hinreichende Menge an Nutzeranforderungen dient als Arbeitsgrundlage für den nächsten Prozessschritt (PDS).	I53	2	Prozess
KoRa.A56	2 UR	Die Ziele, Anforderungen, Wünsche der Nutzer, organisatorische und technische Anforderungen werden gegeneinander abgewogen und werden zur Entwurfs- und Lösungserstellung (PDS) verwendet.	I58	1	AKT.UE.25

Weiter auf der nächsten Seite

NR	UE-Phase	Anforderung	Ref.	Rel.	Abgleich
KoRa.A57	2 UR	Die Nutzeranforderungen beschreiben welche Aktivitäten aus Nutzersicht am System erledigt werden (mit Hilfe des Systems ausgeführt werden können).	I59	1	AKT.UE.33
KoRa.A58	2 UR	Es existiert eine klare Definition darüber, wie Nutzeranforderungen dokumentiert und festgehalten werden (i.S. der inhaltlichen Ausprägung, der Konsistenz, der Verarbeitung).	I62	1	AKT.UE.26 AKT.UE.27
KoRa.A59	2 UR	Innerhalb der Stakeholder (Zielgruppen) werden relevante Nutzergruppen, sowie deren Aufgabenerfordernisse ermittelt und untersucht.	K22	6	AKT.UE.18
KoRa.A60	2 UR	Die Anforderungen der Systemnutzer werden spezifiziert.	K23	6	AKT.UE.19
KoRa.A61	2 UR	Nutzerkriterien zur Messung der Erreichung von operationalen und funktionalen Zielen werden definiert.	K24	6	neu
KoRa.A62	2 UR	Nutzeranforderungen werden im Systemdesign berücksichtigt.	K25	6	erfüllt
KoRa.A63	2 UR	Die Anforderungen an die Usability des Systems ergeben sich (bzw. leiten sich ab) aus den Nutzeranforderungen und der Kontextinformation.	K27	1	AKT.UE.23
KoRa.A64	2 UR	Die abgeleiteten Anforderungen reflektieren den Nutzungskontext und die Situation der Nutzung.	K28	1	AKT.UE.20
KoRa.A65	2 UR	Das erwartete Systemverhalten in Bezug auf die Nutzerinteraktion wird spezifiziert und vereinbart.	K29	6	AKT.UE.29
KoRa.A66	2 UR	Die Nutzeranforderungen an das System werden explizit formuliert.	K30	6	AKT.UE.26
KoRa.A67	2 UR	Die Nutzeranforderungen werden im Hinblick auf Optimierungsmöglichkeiten analysiert.	K31	6	AKT.UE.27
KoRa.A68	2 UR	Messbare Kriterien zur Bewertung des Systems im Nutzungskontext werden definiert und vereinbart.	K32	6	neu
KoRa.A69	2 UR	Die Anforderungen werden Projekt-Stakeholdern zur Weiterverwendung in der Entwicklung und dem Betrieb des Systems weitergegeben.	K33	6	AKT.UE.5
KoRa.A70	2 UR	Es wird dargelegt welche Aktivitäten aus Nutzersicht am System erledigt werden können (mit Hilfe des Systems ausgeführt werden können) und somit die erforderliche Interaktion mit dem System spezifiziert.	K34	1	AKT.UE.28 AKT.UE.29

Weiter auf der nächsten Seite

NR	UE-Phase	Anforderung	Ref.	Rel.	Abgleich
KoRa.A71	2 UR	Es existiert eine Beschreibung und das Wissen darüber, wie Nutzeranforderungen dokumentiert/spezifiziert werden.	K36	1	AKT.UE.6
KoRa.A72	2 UR	Die Erarbeitung der Nutzeranforderungen ist ein iterativer, inkrementeller Prozess.	K40	1	erfüllt
KoRa.A73	3 PDS	Die Konzept- und Lösungserstellung erfolgt iterativ, bis die Anforderungen der Nutzer erfüllt sind und durch Evaluation (Zufriedenheit der Nutzer und der Kunden) bestätigt wird.	I32	5	AKT.UE.43
KoRa.A74	3 PDS	Die Lösung reflektiert die erhobenen Nutzeranforderungen hinreichend.	I33	4	erfüllt
KoRa.A75	3 PDS	Die Lösung reflektiert die Kundenanforderungen hinreichend.	I34	4	erfüllt
KoRa.A76	3 PDS	Die Dokumentation der Lösung ist nachvollziehbar und verständlich für alle Beteiligten.	I35	3	AKT.UE.42
KoRa.A77	3 PDS	Die Personen die an der Lösungserarbeitung beteiligt sind verfügen über hinreichende Erfahrung (etw. „können“ anstatt nur „kennen“; nicht nur theoretisch erlernt sondern praktisch erprobt) in der Umsetzung von Nutzeranforderungen in Konzepte und Designs, unter Berücksichtigung der Kontextinformation.	I36	4	AKT.UE.4
KoRa.A78	3 PDS	Es existieren ausreichend Kapazitäten und Zeit zur Erstellung und Evaluation unterschiedlicher Konzepte, Entwürfe und Lösungen.	I37	3	AKT.UE.8
KoRa.A79	3 PDS	Die Überprüfung der Entwürfe und Lösungen wird mit repräsentativen und geeigneten Nutzern sowie dem Kunden durchgeführt.	I38	3	AKT.UE.43
KoRa.A80	3 PDS	Die erarbeiteten Lösungen dienen als Vorlage für die Implementierung.	I39	3	AKT.UE.44
KoRa.A81	3 PDS	Die Konzept- und Lösungserarbeitung erfolgt in einem interdisziplinären Team, in dem die Rollen des Usability Engineers und des Designers personifiziert sind.	I40	2	Prozess
KoRa.A82	3 PDS	Die Konzept- und Lösungserarbeitung erfolgt in Abstimmung mit/durch Beteiligung der Entwicklung.	I41	1	Prozess
KoRa.A83	3 PDS	Die Lösung erfüllt die 7 Grundsätze der Dialoggestaltung.	I46	1	AKT.UE.30
KoRa.A84	3 PDS	Es werden alternative Lösungen erarbeitet und evaluiert (EoU).	I47	2	AKT.UE.43
KoRa.A85	3 PDS	Die Lösungserstellung erfolgt unter beiderseitigem Einfluss, von Kunden- und Nutzeranforderungen.	I48	1	Prozess

Weiter auf der nächsten Seite

NR	UE-Phase	Anforderung	Ref.	Rel.	Abgleich
KoRa.A86	3 PDS	Teillösungen werden im Gesamtzusammenhang, bezogen auf den Kontext, entwickelt und evaluiert (EoU).	I49	1	AKT.UE.41
KoRa.A87	3 PDS	Die Evaluation der Lösung erfolgt gegen die erhobenen Nutzeranforderungen.	I54	1	AKT.UE.47
KoRa.A88	3 PDS	Analysten sind aktiv am Designprozess (PDS) beteiligt.	I9	1	Prozess
KoRa.A89	3 PDS	Nutzungsprobleme und Usability-Kriterien werden bei Entscheidungen zu Design-Alternativen berücksichtigt.	K42	6	AKT.UE.40
KoRa.A90	3 PDS	Usability is traded-off against other design criteria	K43	6	AKT.UE.41
KoRa.A91	3 PDS	Zu allen relevanten Nutzeraspekten (z. B. Tätigkeiten, Rollen, Dokumentation, Personal) existiert eine Lösung.	K44	6	Prozess
KoRa.A92	3 PDS	Nutzer-Input, sowohl direkt als auch als Feedback aus Evaluationen, wird im Design berücksichtigt.	K45	6	AKT.UE.43
KoRa.A93	3 PDS	Die Konzepte und Lösungen werden für die spätere Verwendung (ggf. Nachfolgeprojekte) dokumentiert.	K49	1	AKT.UE.6
KoRa.A94	3 PDS	Die Lösungserstellung ist eng verzahnt mit der Tätigkeit der Evaluation (EoU).	K51	1	Prozess
KoRa.A95	3 PDS	Die Lösungserstellung ist eine iterative Tätigkeit.	K52	1	AKT.UE.8
KoRa.A96	3 PDS	Es erfolgt eine Funktionsteilung zwischen Mensch, Maschine und Organisation basierend jeweils auf der bestmöglichen Eignung für die gegebene Funktion.	K53	6	AKT.UE.33
KoRa.A97	3 PDS	Ein praktisches Modell der Nutzeraufgaben wird entwickelt, basierend auf den Anforderungen, dem Nutzungskontext, der Funktionsteilung sowie anderen Rahmenbedingungen.	K54	6	AKT.UE.28 AKT.UE.32
KoRa.A98	3 PDS	Für die nutzerrelevanten Anteile des Systems werden Lösungen (Konzepte, Designs) produziert, die die Anforderungen, den Nutzungskontext sowie das ergonomische Wissen berücksichtigen.	K55	6	AKT.UE.40 AKT.UE.41
KoRa.A99	3 PDS	Die zukünftige Benutzung/Bedienung des Systems wird beschrieben.	K56	6	AKT.UE.37
KoRa.A100	3 PDS	Das Design sowie Sicherheitsaspekte werden basierend auf Evaluations-Feedback weiterentwickelt.	K57	6	AKT.UE.43
KoRa.A101	3 PDS	Es ist sichergestellt das das Wissen über die Lösungserstellung und die Lösung hinreichend kommuniziert wird.	K58	1	AKT.UE.44
KoRa.A102	3 PDS	Es erfolgt eine Interaktionsanalyse, bei der klar dargelegt wird was aus Nutzersicht zu tun ist und was aus Systemsicht zu tun ist.	K61	1	AKT.UE.43 AKT.UE.44

Weiter auf der nächsten Seite

NR	UE-Phase	Anforderung	Ref.	Rel.	Abgleich
KoRa.A103	4 EoU	Die Evaluation der (Teil-)Lösungen erfolgt iterativ, bis alle feststellbaren, signifikanten Nutzungsprobleme erhoben sind und eine Zufriedenheit der Nutzer gewährleistet ist.	I50	4	AKT.UE.8
KoRa.A104	4 EoU	Die Ergebnisse der Evaluation fließen in den Prozess zurück und eine Bearbeitung ist sichergestellt.	I51	4	AKT.UE.59
KoRa.A105	4 EoU	Die Evaluation erfolgt auf Grundlage der erhobenen Kontextinformation (Nutzungsszenarien) und zusätzlich in Bezug auf die Nutzeranforderungen und Kundenbelange.	I55	2	erfüllt
KoRa.A106	4 EoU	Die formative Evaluation ist Bestandteil des Prozesses und dient der Überprüfung der (Nutzer-)Ziele.	I60	1	AKT.UE.46
KoRa.A107	4 EoU	Die Evaluation wird mit echten, repräsentativen Nutzern durchgeführt.	I61	1	AKT.UE.48 AKT.UE.54
KoRa.A108	4 EoU	Basierend auf entwicklungsbegleitender Evaluation werden Gestaltungshinweise, neue Risiken, sowie Probleme erhoben.	K62	4	AKT.UE.52
KoRa.A109	4 EoU	Durch abschließende Evaluation wird die Erfüllung von Nutzeranforderungen nachgewiesen/demonstriert.	K63	4	AKT.UE.47
KoRa.A110	4 EoU	Die (Entwicklungs-)Organisation hat ausreichende Information, auf deren Basis sie Entscheidungen zu HCI Problemen treffen kann.	K64	4	Prozess
KoRa.A111	4 EoU	Die Lösung/ das Design wird mit Nutzern evaluiert.	K65	4	AKT.UE.48 AKT.UE.54
KoRa.A112	4 EoU	Durch formative Evaluation wird überprüft, ob die Ziele erreicht wurden oder ob nicht.	K66	1	AKT.UE.46
KoRa.A113	4 EoU	Die Evaluation wird mit tatsächlichen Nutzern durchgeführt.	K67	1	AKT.UE.48 AKT.UE.54
KoRa.A114	4 EoU	Die Evaluation wird anhand der vorliegenden Lösungen (Prototypen) und schließlich mit dem fertigen Produkt durchgeführt.	K68	1	Prozess
KoRa.A115	4 EoU	Die Evaluation ist im gesamten Prozess verankert und wird zur Überprüfung aller Ergebnisse verwendet.	K69	1	Prozess
KoRa.A116	4 EoU	Es existieren (neben Nutzer- Nutzungs- und technischen Anforderungen) auch Anforderungen an das Design. Auch diese müssen evaluiert werden.	K71	1	AKT.UE.41
KoRa.A117	4 EoU	Die Evaluation wird geplant.	K74	4	AKT.UE.50
KoRa.A118	4 EoU	Die Rahmenbedingungen der Evaluation werden identifiziert und betrachtet.	K75	4	AKT.UE.49 AKT.UE.50

Weiter auf der nächsten Seite

NR	UE-Phase	Anforderung	Ref.	Rel.	Abgleich
KoRa.A119	4 EoU	Es wird sichergestellt, dass das System reif für eine Evaluation ist.	K76	4	AKT.UE.57
KoRa.A120	4 EoU	Die Evaluation und Auswertung werden entsprechend dem Evaluationsplan durchgeführt: a) Frühe Prototypen werden evaluiert um die Nutzeranforderungen an das System herauszufinden. b) Prototypen werden evaluiert, um das Konzept und Design zu optimieren c) Das System wird evaluiert um zu überprüfen, ob alle Stakeholder- und organisationalen Anforderungen erfüllt sind. d) Das System wird evaluiert um sicherzustellen, dass das erforderliche Vorgehen angewandt wurde. e) Das System wird in der realen Nutzungsumgebung evaluiert um sicherzustellen, dass es dort auch weiterhin die Anforderungen der Nutzer sowie der Organisation erfüllen.	K77	4	AKT.UE.50 AKT.UE.46 AKT.UE.47 AKT.UE.57 AKT.UE.56 AKT.UE.58
KoRa.A121	4 EoU	Basierend auf dem Verständnis über die Evaluationsergebnisse werden weitere Systementscheidungen getroffen.	K78	4	AKT.UE.52
KoRa.A122	4 EoU	Die Überprüfung, ob das System reif für die Evaluation ist, erfolgt sowohl durch Experten der Usability, aber auch durch Experten der Umsetzung.	K79	1	AKT.UE.57
KoRa.A123	4 EoU	Neben der Überprüfung, ob das System fit für die Evaluation ist, ist gewährleistet das eine entsprechende Testumgebung bereit steht.	K81	1	AKT.UE.50
KoRa.A124	4 EoU	Es wird überprüft ob die erhobenen/erarbeiteten Anforderungen erfüllt wurden.	K82	1	AKT.UE.47

Tabelle E.1.: Konformitäts- und Rahmenanforderungen nach Nebe (2009)

F. Expertenbefragung

Ergebnisse der dimensionalen Analyse des Untersuchungsgegenstandes, der Gewichtung der einzelnen Aktivitäten und Artefakte durch die Experten, sowie die Aussagen der Experten auf Basis der Fragen.

F.1. Dimensionale Analyse

Ergebnisse der in Kapitel 4.2 referenzierten dimensionalen Analyse.

Erläuterungen zu Tabelle F.1:

- Aspekte: Aspekte, anhand derer das Integrationsmodell bewertet werden soll
- Dimensionen: Ausprägung, anhand derer der Aspekt operationalisiert werden soll
- Fragegegenstand: Beschreibt den zu hinterfragenden Gegenstand
- Frage: Ausformulierte Frage auf Basis des Fragegegenstandes (semantisch analysiert)
- Nr: Reihenfolge, in der die Fragen gestellt werden sollen

Aspekte	Dimensionen	Fragegegenstand	Frage	Nr
Akzeptanz	Usability	Einschätzung über die Relevanz des Aspektes „Usability“ in Unternehmen in Hinblick auf deren strategische Ausrichtung	Wie schätzen Sie die Relevanz von Usability in den Unternehmen ein, bezogen auf deren strategische Ausrichtung? (<i>Likert Skala: sehr hoch, hoch, neutral, niedrig, sehr niedrig</i>)	1
		Einschätzung über die Relevanz des Aspektes „Usability“ in Unternehmen in Hinblick auf deren operatives Geschäft	Wie schätzen Sie die Relevanz von Usability in den Unternehmen ein, bezogen auf das operative Geschäft? (<i>Likert Skala: sehr hoch, hoch, neutral, niedrig, sehr niedrig</i>)	2

Weiter auf der nächsten Seite

Aspekte	Dimensionen	Fragegegenstand	Frage	Nr
		Einschätzung über die Relevanz des Aspektes „Usability“ in Unternehmen in Hinblick auf deren Unternehmenserfolg	Wie schätzen Sie die Relevanz von Usability in den Unternehmen ein, bezogen auf deren Erfolg? (<i>Likert Skala: sehr hoch, hoch, neutral, niedrig, sehr niedrig</i>)	3
	Human-Centred Design	Einschätzung, inwieweit benutzerorientierte Gestaltungsansätze in der Praxis Verwendung finden	Wie hoch schätzen Sie die Verbreitung von Entwicklungsansätzen in der Praxis ein, deren Fokus auf den Benutzern liegt? (<i>Likert Skala: sehr hoch, hoch, neutral, niedrig, sehr niedrig</i>)	4
Vollständigkeit	Phasen	Einschätzung, ob die vier Phasen der ISO 9241-210 vollständig sind für die Durchführung einer benutzerorientierten Entwicklung	Wie beurteilen Sie die Vollständigkeit der in der ISO 9241-210 definierten Phasen zur Durchführung einer benutzerorientierten Entwicklung?	5
		Einschätzung zur zeitlichen Anordnung der vier Phasen der ISO 9241-210	Wie beurteilen Sie die zeitliche Anordnung der Phasen in der ISO 9241-210?	6
		Einschätzung zur Relevanz der einzelnen Phasen der ISO 9241-210	Bewerten Sie die in der folgenden Liste aufgeführten Phasen der ISO 9241-210. (<i>Likert Skala: sehr bedeutend, bedeutend, neutral, weniger bedeutend, unbedeutend</i>)	7
		Einschätzung, anhand welcher Indikatoren eine Phase beendet wird	Welche Indikatoren kennzeichnen das Ende einer Phase [...]?	10
		Einschätzung, auf Grund welcher Kriterien eine Phase als erfolgreich angesehen wird	Anhand welcher Kriterien lässt sich der Erfolg der Phase [...] messen?	9
	Aktivitäten	Einschätzung, welche Aktivitäten in einer Phase durchgeführt werden	Aus welchen Aktivitäten setzt sich die Phase [...] zusammen?	11
		Einschätzung zur Aktivitätenliste auf Basis des Modells	Betrachten Sie folgende Liste von Aktivitäten und ergänzen Sie diese falls notwendig.	12
		Einschätzung zur Relevanz der einzelnen Aktivitäten auf Basis des Modells	Bewerten Sie die in der folgenden Liste aufgeführten Aktivitäten. (<i>Likert Skala: sehr bedeutend, bedeutend, neutral, weniger bedeutend, unbedeutend</i>)	13

Weiter auf der nächsten Seite

Aspekte	Dimensionen	Fragegegenstand	Frage	Nr
	Artefakte	Einschätzung, in welcher Dokumentationsart die Arbeitsergebnisse festgehalten werden	In welcher Form werden die Ergebnisse der Phase [...] dokumentiert?	15
		Einschätzung zur Artefaktliste auf Basis des Modells	Betrachten Sie folgende Liste von Dokumentationsformen und ergänzen Sie diese falls notwendig.	16
		Einschätzung zur Relevanz der einzelnen Artefakte auf Basis des Modells	Bewerten Sie die in der folgenden Liste aufgeführten Dokumentationsformen. (<i>Likert Skala: sehr bedeutend, bedeutend, neutral, weniger bedeutend, unbedeutend</i>)	17
Abhängigkeiten	Phasen	Einschätzung, wie die einzelnen Phasen in Wechselwirkung zueinander stehen	Welche Wechselwirkungen existieren zwischen den vier Phasen der ISO 9241-210?	8
		Einschätzung, welche Eingangsvoraussetzungen (Artefakte, Informationen, Wissen) für den Beginn eine Phase notwendig sind	Welche Informationen sind notwendig für den Beginn der Phase [...]?	14
These 1	Generische Phasen	Einschätzung, inwieweit die vier generischen Phasen der ISO 9241-210 für den allgemeinen Lebenszyklus eines Systems akzeptiert werden	Würden Sie behaupten, dass die vier generischen Phasen der ISO 9241-210 auch in der Disziplin des Software Engineering für den übergeordneten Lebenszyklus eines Systems akzeptiert werden?	23
These 2	Artefakt-lebenszyklus	Einschätzung, wie die einzelnen Phasen in Wechselwirkung zueinander stehen und welche Eingangsvoraussetzungen für den Beginn eine Phase notwendig sind	s.o. (<i>Thema: Abhängigkeiten</i>)	8, 10, 14
These 3	Nutzungskontext	Einschätzung über die Wichtigkeit einer genauen Analyse des Nutzungskontextes für das Verständnis des Problemraums und deren Auswirkung auf die Qualität der Lösung	Wie schätzen Sie die Bedeutung einer Analyse des Nutzungskontextes in Bezug auf das Verständnis des Problemraumes ein?	21

Weiter auf der nächsten Seite

Aspekte	Dimensionen	Fragegegenstand	Frage	Nr
			Wie beurteilen Sie den Zusammenhang zwischen dem genauen Verständnis des Problemraumes und der Qualität der Lösung?	22
These 4	Artefakte	Einschätzung zu einer gemeinsamen (UE und SE) Anforderungsspezifikation als Basis für den Gestaltungs- und Entwicklungsprozess	Würden Sie behaupten, dass UE und SE durch eine gemeinsam angefertigte Anforderungsspezifikation die Qualität der daraus resultierenden Ergebnisse steigern können?	27
These 5	Evaluation	Einschätzung, ob Evaluation eine Aktivität am Ende jeder Iteration ist oder bereits während jeder Phase durchgeführt werden sollte	Würden Sie behaupten, dass eine Evaluation nicht nur zum Abschluss einer Iteration, sondern bereits nach jeder Phase durchgeführt werden sollte?	18
These 6	Evaluations-techniken	Einschätzung, inwieweit die ersten drei Phasen der ISO 9241-210 und die dort erstellten Artefakte zur Anwendung gewisser Evaluationstechniken benötigt werden	Würden Sie behaupten, dass die Beschreibung des Nutzungskontextes und die Anforderungsspezifikation essentiell für die Durchführung gewisser Evaluationstechniken (wie z.B. „Cognitive Walkthrough“, „Think Aloud“ oder „Usability Testing“) sind?	19
			Wie beurteilen Sie das Ergebnis der Evaluation eines existierenden Produktes, wenn dazu keine Beschreibung des Nutzungskontextes und keine Anforderungsspezifikation vorliegt?	20
These 7	Aktivitäten	Einschätzung, ob eine Schnittmenge von Aktivitäten zwischen dem UE und SE existiert	Würden Sie behaupten, dass ohne die Integration das UE und SE unabhängig voneinander Aktivitäten durchführen die sich gleichen?	24
These 8	Artefakte	Einschätzung zu Anknüpfungspunkten zwischen den Disziplinen und einer damit verbundenen gegenseitigen Qualitätssteigerung	Würden Sie behaupten, dass durch einen Informationsaustausch an im Prozess fest definierten Punkten die Qualität der jeweils von UE und SE erstellten Ergebnisse gesteigert wird?	25

Weiter auf der nächsten Seite

Aspekte	Dimensionen	Fragegegenstand	Frage	Nr
			Wie sehen Ihrer Meinung nach Anknüpfungspunkte zwischen den Disziplinen aus?	26

Tabelle F.1.: Dimensionale Analyse des Untersuchungsgegenstandes

F.2. Gewichtung der Aktivitäten und Artefakte

Ergebnisse der Gewichtung von Aktivitäten und Artefakten durch die Experten, wie in Kapitel 4.2 beschrieben.

Erläuterungen zu Tabellen F.2 und F.4:

- Referenz: Eindeutige Bezeichnung einer Aktivität bzw. eines Artefakts
- UE-Phase: Einordnung einer Aktivität bzw. eines Artefakts in eine Phase des UE: „Organisatorische Vorbereitungen“, „Verstehen und Beschreiben des Nutzungskontext“ (VBN), „Festlegen von Nutzungsanforderungen“ (FNA), „Entwerfen von Gestaltungslösungen“ (EGL), „Beurteilen von Gestaltungslösungen“ (BGL), „System erfüllt festgelegte Anforderungen“ (SAF)
- Phase: Beschreibung der Phase
- Aktivität: Beschreibung der Aktivität
- Artefakt: Beschreibung der Artefakts
- Likert Skala: Fünfstufige Gewichtung von sehr bedeutend bis unbedeutend
- Relevanz: Wenn die Skala mit vier („sehr bedeutend“) bis null („unbedeutend“) beziffert wird, lässt sich die Relevanz berechnen aus der Summe, der jeweiligen Stimmenanzahl multipliziert mit dem jeweiligen Skalawert, dividiert durch die Anzahl der Experten:

$$\frac{\Sigma(\text{Stimmenanzahl} * \text{Skalawert})}{\text{Expertenanzahl}}$$

F.2.1. Phasen

Nr	Phase	sehr bedeutend	bedeutend	neutral	weniger bedeutend	unbedeutend	RELEVANZ
0	Organisatorische Vorbereitungen	1	1				3,5
1	Verstehen und Beschreiben des Nutzungskontext	1	1				3,5
2	Festlegen von Nutzungsanforderungen	2					4
3	Entwerfen von Gestaltungslösungen	1	1				3,5
4	Beurteilen von Gestaltungslösungen		2				3

Tabelle F.2.: Gewichtung der Phasen

F.2.2. Aktivitäten

Referenz	UE-Phase	Aktivität	sehr bedeutend	bedeutend	neutral	weniger bedeutend	unbedeutend	RELEVANZ
AKT.UE.1		Festlegen der Notwendigkeit einer benutzerorientierten Gestaltung			2			2
AKT.UE.2		Identifizieren von adäquaten Methoden und Ressourcen	1	1				3,5
AKT.UE.3		Definieren von Prozeduren zur Integration von Aktivitäten und Artefakten mit anderen Systementwicklungsaktivitäten	2					4
AKT.UE.4		Identifizieren der für die benutzerorientierten Gestaltungsaktivitäten zuständigen Personen und Organisationen, sowie deren Fähigkeiten und Perspektiven		2				3
AKT.UE.5		Entwickeln von effektiven Prozeduren zur Etablierung von Rückmeldungen und Kommunikation über benutzerorientierte Gestaltungsaktivitäten	1	1				3,5

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität	sehr bedeutend	bedeutend	neutral	weniger bedeutend	unbedeutend	RELEVANZ
AKT.UE.6		Entwickeln von effektiven Methoden zur Dokumentation der Erkenntnisse		2				3
AKT.UE.7		Einigen über angemessene Meilensteine für benutzerorientierte Aktivitäten, integriert in den übergeordneten Design- und Entwicklungsprozess	1	1				3,5
AKT.UE.8		Einigen über entsprechende Zeitskalen für Iterationen, den Gebrauch von Rückmeldungen und mögliche Gestaltungsänderungen, eingegliedert in den Projektplan	2					4
AKT.UE.9		Definieren einer Strategie und eines Plans zur Einbindung von Benutzer		2				3
AKT.UE.10	VBN	Eingrenzen des Nutzungskontextes (Scope)	2					4
AKT.UE.11	VBN	Identifizieren von Benutzern und anderen Stakeholdern	2					4
AKT.UE.12	VBN	Identifizieren der Charakteristiken von Benutzern und Benutzergruppen		1	1			2,5
AKT.UE.13	VBN	Identifizieren der Aufgaben der Benutzer und damit verbundene Ziele	2					4
AKT.UE.14	VBN	Identifizieren der Umgebung(en) des Systems (organisatorisch, sozial, kulturell, physisch)	1	1				3,5
AKT.UE.15	VBN	Beeinflussen bzw. Abändern der Aufgaben und organisatorischen Rahmenbedingungen		2				3
AKT.UE.16	VBN	Einschätzen von Auswirkungen auf die Gesellschaft und das Umfeld durch menschliche Fehler beim Gebrauch des Systems		2				3
AKT.UE.17	VBN	Einschätzen von Risiken für die physische und psychische Gesundheit und das Wohlbefinden der Benutzer selbst	1	1				3,5
KoRa.A9	VBN	Sicherstellen, dass die Kontextanalyse im richtigen, originären Kontext durchgeführt wird	1	1				3,5
KoRa.A11	VBN	Validieren der Kontextinformationen mit repräsentativen Nutzern und Kunden		1	1			2,5
KoRa.A12	VBN	Abbildern der Zusammenhänge zwischen Nutzern, Aufgaben, Zielen, Charakteristiken der Nutzeraufgaben und deren Umgebung, sowie den zugehörigen Rahmenbedingungen	1	1				3,5

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität	sehr bedeutend	bedeutend	neutral	weniger bedeutend	unbedeutend	RELEVANZ
KoRa.A13	VBN	Sicherstellen, dass Kontextinformationen (für Dritte) verständlich und nachvollziehbar formuliert sind	1		1			3
KoRa.A18	VBN	Sicherstellen, dass Kontextinformationen Fakten abbilden und keine Interpretation der Situation darstellen	1		1			3
KoRa.A21	VBN	Abbilden der Kontextinformationen auf organisatorische bzw. Kundenziele		1	1			2,5
KoRa.A24	VBN	Sicherstellen, dass die Kontextinformationen die Sichtweise der Nutzer und keine technische Sichtweise repräsentiert	1		1			3
KoRa.A37	VBN	Demonstrieren der Ergebnisse für Projekt-Stakeholder			2			2
AKT.UE.18	FNA	Identifizieren der Erfordernisse (Needs) von Benutzern und anderen Stakeholdern	1	1				3,5
AKT.UE.19	FNA	Ableiten von Nutzungsanforderungen	2					4
AKT.UE.20	FNA	Ableiten von Anforderungen aus dem Nutzungskontext	1		1			3
AKT.UE.21	FNA	Ableiten von Anforderungen aus den Erfordernissen (Needs) im Nutzungskontext	1	1				3,5
AKT.UE.22	FNA	Ableiten von Anforderungen aus relevanten Kenntnissen über Ergonomie und Benutzungsschnittstellen, Standards und Richtlinien	1	1				3,5
AKT.UE.23	FNA	Aufstellen von messbaren Kriterien der Gebrauchstauglichkeit	1		1			3
AKT.UE.24	FNA	Ableiten von organisatorischen Anforderungen, die den Benutzer direkt betreffen		2				3
AKT.UE.25	FNA	Beseitigen von Zielkonflikten zwischen Nutzungsanforderungen			2			2
AKT.UE.26	FNA	Dokumentieren von Begründungen, Faktoren und Gewichtungen der Mensch-System Belange			1		1	1
AKT.UE.27	FNA	Sicherstellen der Qualität der Nutzungsanforderungsspezifikation (überprüfbar formuliert, verifiziert durch Stakeholder, konsistent, aktualisiert)	2					4
KoRa.A43	FNA	Sicherstellen, dass die Nutzungsanforderungen systemneutral (unabhängig einer Lösung) formuliert sind	1	1				3,5
KoRa.A61	FNA	Definieren von Kriterien zur Messung der Erreichung von Anforderungen		1		1		2
KoRa.A68	FNA	Definieren und vereinbaren von messbaren Kriterien zur Bewertung des Systems im Nutzungskontext		1		1		2

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität	sehr bedeutend	bedeutend	neutral	weniger bedeutend	unbedeutend	RELEVANZ
AKT.UE.28	EGL	Gestalten von Benutzeraufgaben	1				1	2
AKT.UE.29	EGL	Gestalten von Benutzer-System-Interaktion unter Berücksichtigung von Nutzungserlebnis (UX)		1	1			2,5
AKT.UE.30	EGL	Berücksichtigen von Gestaltungsprinzipien (siehe ISO 9241-110: Aufgabenangemessenheit, Selbstbeschreibungsfähigkeit, Erwartungskonformität, Lernförderlichkeit, Steuerbarkeit, Fehlertoleranz, Individualisierbarkeit)	1			1		2
AKT.UE.31	EGL	Aufstellen von Gestaltungsentscheidungen auf konzeptioneller Ebene		1	1			2,5
AKT.UE.32	EGL	Identifizieren von Aufgaben und Teilaufgaben	1		1			3
AKT.UE.33	EGL	Zuordnen von Aufgaben und Teilaufgaben zu Benutzern und anderen Systemkomponenten		2				3
AKT.UE.34	EGL	Identifizieren von benötigten Interaktionsobjekten	2					4
AKT.UE.35	EGL	Identifizieren und Auswählen von angemessenen Dialogtechniken (siehe ISO 9241-12 bis ISO 9241-17)	1	1				3,5
AKT.UE.36	EGL	Gestalten von Sequenzen der Interaktion	1	1				3,5
AKT.UE.37	EGL	Gestalten der Informationsarchitektur der Benutzungsschnittstelle	2					4
AKT.UE.38	EGL	Gestalten der Systeminfrastruktur			1		1	1
AKT.UE.39	EGL	Gestalten der Benutzungsschnittstelle unter Berücksichtigung von Nutzungserlebnis (UX)		2				3
AKT.UE.40	EGL	Berücksichtigen von relevanten Kenntnissen über Ergonomie und Benutzungsschnittstellen, Standards und Richtlinien		2				3
AKT.UE.41	EGL	Berücksichtigen von internen Gestaltungsrichtlinien für Benutzungsschnittstellen, Produktkenntnissen, Kenntnissen über die Benutzer und weiteren Aspekten des Nutzungskontextes (z.B. UX und Stereotypen)	1	1				3,5
AKT.UE.42	EGL	Konkretisieren von Gestaltungslösungen (mittels Szenarien, Simulationen, Modellen, Prototypen)	2					4
AKT.UE.43	EGL	Modifizieren von Gestaltungslösungen auf Basis von benutzerorientierter Evaluation und Rückmeldungen	1	1				3,5

Weiter auf der nächsten Seite

Referenz	UE-Phase	Aktivität	sehr bedeutend	bedeutend	neutral	weniger bedeutend	unbedeutend	RELEVANZ
AKT.UE.44	EGL	Kommunizieren der Gestaltungslösung zu Verantwortlichen der Implementierung	1	1				3,5
AKT.UE.45	BGL	Sammeln von neuen Informationen über die Erfordernisse (Needs) der Benutzer	1		1			3
AKT.UE.46	BGL	Bereitstellen von Rückmeldungen über Stärken und Schwächen der Gestaltungslösung aus Sicht der Benutzer (Formative Evaluation)	2					4
AKT.UE.47	BGL	Beurteilen, ob die Benutzeranforderungen erreicht wurden (Konformitätsprüfung)	1	1				3,5
AKT.UE.48	BGL	Durchführen einer benutzerorientierten Evaluation	1	1				3,5
AKT.UE.49	BGL	Bereitstellen von Ressourcen	1	1				3,5
AKT.UE.50	BGL	Planen einer benutzerorientierten Evaluation	2					4
AKT.UE.51	BGL	Durchführen einer umfassenden Prüfung, um Ergebnisse über das System als Ganzes zu erhalten	1	1				3,5
AKT.UE.52	BGL	Ergebnisse analysieren und erforderliche Anpassung der Lösungen entwickeln	1		1			3
AKT.UE.53	BGL	Kommunizieren der Ergebnisse	1	1				3,5
AKT.UE.54	BGL	Testen mit Benutzern	2					4
AKT.UE.55	BGL	Beurteilen, ob Nutzungsanforderungen erreicht wurden		1	1			2,5
AKT.UE.56	BGL	Durchführen einer Feldvalidierung	1		1			3
AKT.UE.57	BGL	Durchführen einer inspektionsbasierten Prüfung und Bewertung (Expertentest)		1	1			2,5
AKT.UE.58	BGL	Durchführen von Langzeitbeobachtungen		1	1			2,5
AKT.UE.59	BGL	Sicherstellen, dass gewonnene Erkenntnisse und Daten angewandt werden bzw. Einfluss nehmen	2					4
KoRa.A17	BGL	Berücksichtigen der Erfordernisse im Kontext als Validierungsgrundlage für die Lösung	1	1				3,5

Tabelle F.3.: Gewichtung der Aktivitäten

F.2.3. Artefakte

Referenz	UE-Phase	Artefakt	sehr bedeutend	bedeutend	neutral	weniger bedeutend	unbedeutend	RELEVANZ
ARF.UE.1		Plan der benutzerorientierten Gestaltung	2					4
ARF.UE.2		Nachweis der Kommunikation		1	1			2,5
ARF.UE.3		Ziele der benutzerorientierten Gestaltung	2					4
ARF.UE.4	VBN	Beschreibung des gegebenen Nutzungskontextes	1	1				3,5
ARF.UE.5	VBN	Beschreibung des festgelegten Nutzungskontextes für das Design		1		1		2
ARF.UE.6	FNA	Beschreibung der Erfordernisse der Benutzer	1	1				3,5
ARF.UE.7	FNA	Ziele der Gebrauchstauglichkeit	1		1			3
ARF.UE.8	FNA	Messbare Kriterien (Gebrauchstauglichkeit, Zufriedenheit)	1		1			3
ARF.UE.9	FNA	Spezifikation der Nutzungsanforderungen	1	1				3,5
ARF.UE.10	FNA	Informationen über Ergonomie	1	1				3,5
ARF.UE.11	FNA	Informationen über Benutzungsschnittstellengestaltung	1			1		2,5
ARF.UE.12	FNA	Standards / Normen		2				3
ARF.UE.13	FNA	Richtlinien (u.a. Styleguides des Unternehmens)	1	1				3,5
ARF.UE.14	EGL	Spezifikation der Benutzerinteraktion (einschließlich Aufgaben und Teilaufgaben)	2					4
ARF.UE.15	EGL	Informationen über das Nutzungserlebnis			1	1		1,5
ARF.UE.16	EGL	Designprinzipien (u.a. ISO 9241-110)	1		1			3
ARF.UE.17	EGL	Dialogtechniken (ISO 9241-12 bis ISO 9241-17)		1	1			2,5
ARF.UE.18	EGL	Gestaltungsrichtlinien		2				3
ARF.UE.19	EGL	Produktkenntnisse	1		1			3
ARF.UE.20	EGL	Marketinginformationen (Vorlieben, Abneigungen, Wünsche)		2				3
ARF.UE.21	EGL	Spezifikation der Benutzungsschnittstelle	2					4
ARF.UE.22	EGL	Umsetzung der Benutzungsschnittstelle in einem Prototyp	1	1				3,5
ARF.UE.23	BGL	Evaluationsplan		2				3
ARF.UE.24	BGL	Ergebnisse der Evaluation	2					4
ARF.UE.25	BGL	Ergebnisse der Konformitätsprüfung	2					4
ARF.UE.26	BGL	Ergebnisse der Langzeitbeobachtung (z.B. Rückmeldungen)		1	1			2,5

Weiter auf der nächsten Seite

Referenz	UE-Phase	Artefakt	sehr bedeutend	bedeutend	neutral	weniger bedeutend	unbedeutend	RELEVANZ
ARF.UE.27	BGL	Rückmeldungen		1	1			2,5
ARF.UE.28	BGL	Leistungskriterien	1		1			3
ARF.UE.29	BGL	Gesundheitsberichte (medizinisch)			1		1	1
ARF.UE.30	BGL	Bericht über Zielerreichung	1		1			3
ARF.UE.31	BGL	Bericht über die Behandlung von Nichtkonformitäten	1				1	2

Tabelle F.4.: Gewichtung der Artefakte

F.3. Aussagen der Experten

Aussagen der Experten auf Basis der Fragen anhand des Befragungsplans in Kapitel 4.2.

Erläuterungen zu Tabellen F.5:

- Frage: Referenz zur jeweiligen Frage
- Experte: Referenz zum jeweiligen Experten
- Aussage: Jeweilige Aussage der Experten zur Frage

Frage	Experte	Aussage
F.1	E1	In Software-herstellenden Unternehmen ist Usability, besonders seit Aufnahme des Begriffes User Experience ins Vokabular, faktisch ein großer strategischer Pforten. In der Software-anwendenden Industrie ist die Relevanz von Usability nicht eindeutig festlegbar. Unter dem Aspekt der Effizienzsteigerung und Konsensbildung in Entwicklungsprojekten werden diese Unternehmen jedoch irgendwann bei Usability ankommen. Usability wird relativ wenig von den Unternehmen nach außen kommuniziert, ist aber durchaus Vorstandsthema.
	E2	Die Relevanz ist stark domänenabhängig. Je näher das Produkt am Endnutzer ist, desto mehr spielt Usability auf der strategischen Ebene eine Bedeutung in Hinblick auf Effizienzsteigerung, Kundengewinnung, Kundenbindung.

Weiter auf der nächsten Seite

Frage	Experte	Aussage
F.2	E1	Usability hat eine hohe Relevanz im operativen Geschäft, sobald IT-Unterstützung notwendig ist, um zum Beispiel den asynchrone Austausch von Informationen im Unternehmen zu gewährleisten. In Produktionsbetrieb spielt Usability auf operativer Ebene eine große Rolle im Problemmanagement (Bsp. Maschinenausfall)
	E2	Nach wie vor ist Usability nicht selbstverständlich und wird teilweise nur am Rande betrachtet.
F.3	E1	In der Welt des Wettbewerbs ist die Relevanz von Usability für den Geschäftserfolg sehr hoch.
	E2	In der Branche des Unternehmens ist Usability eher unwichtig, spielt aber eine sehr große Rolle bei den vertriebenen Endgeräten.
F.4	E1	Die Unternehmen versuchen an den Nutzer zu denken, führen aber keinen systematischen Ansatz durch.
	E2	Von der Gesamtzahl aller Projekte im Unternehmen werden benutzerorientierte Ansätze in 25% der Projekte durchgeführt. Bei einem Großteil der verbleibenden 75% handelt es sich um Projekte, bei denen Usability keine Relevanz hat oder der Personalbedarf nicht abgedeckt ist.
F.5	E1	In Bezug auf Usability sind die Phasen (Activities) ausreichend benannt. Die inhaltlichen Ausgestaltung muss noch wesentlich genauer durchgeführt werden, ist aber nur soweit möglich, wie ein Konsens unter den Experten hergestellt werden kann.
	E2	Wird persönlich nicht als Phasenmodell interpretiert, da Aktivitäten nicht unbedingt sequenziell, sondern vielmehr parallel ablaufen. Generell werden Usability Tätigkeiten abgedeckt. Auf das Projekt bezogen, fehlen die Entwicklungsbegleitung, die Überprüfung, ob Styleguides eingehalten werden und Aspekte wie Vergleichbarkeit, Konsistenz oder Integrierbarkeit. Dies müsste jedoch eigentlich eine Ebene darüber in der Organisation von Projekten angesiedelt werden.
F.6	E1	Die zeitliche Anordnung der Phasen wird offen gelassen. Die Abbildung in der Norm stellt lediglich die Abhängigkeiten (Dependencies) dar. Es ist kein Prozessmodell. Der Einstieg ist abhängig von der Art des Projektes.
	E2	Es sind keine Phasen. Phasen sind eher versetzte Sinuskurven (Wellenmodell). Z.B. viele neue Anforderungen können aus den Erfahrungen des alten Systems gewonnen werden. Phasen sind über den gesamten Projektverlauf ständig aktiv. Komplett entkoppelt.
F.7		<i>siehe Anhang F.2 - Tabelle F.2</i>
F.8	E1	In einem Projekt wird faktisch immer mit den Ergebnissen einer Vorphase gearbeitet, auch wenn diese nicht durchgeführt wurde. In diesem Fall werden Antizipationen getroffen. Je weniger Budget in der Analyse ausgegeben wurde, desto mehr muss im Abschluss getestet werden.
	E2	Es besteht ein logischer Fluss in der aktuellen Abbildung im Standard. Bei Innovationsprojekten ist jedoch die Lösungsidee Innovationstreiber und nicht das Verständnis des Problems.

Weiter auf der nächsten Seite

Frage	Experte	Aussage
		Phasen wechseln sich auch untereinander ab. Evaluation (Validierung) schwebt über allen anderen.
F.9	E1	VBN: Ausreichende valide Daten des Nutzungskontextes zur Ableitung von Nutzeranforderungen. FBN: Vollständige valide Anforderungen. EGL: Gestaltungslösungen erfüllen die Anforderungen. BGL: Interpretierbare Evaluationsergebnisse, die zur Verbesserung der Gestaltungslösung beitragen.
	E2	<i>Konsens mit E1</i>
F.10	E1	VBN: Es gibt keine zu klärenden Fragen mehr über den Kontext. FBN: Es können vorerst keine Anforderungen mehr abgeleitet werden. EGL: Es konnten alternative Gestaltungslösungen erstellt werden. BGL: Ergebnisse der Evaluation mit den Nutzern liegen vor und Empfehlungen wurden formuliert.
	E2	<i>Konsens mit E1</i>
F.11, 12	E1	<i>Ergänzende, im Modell noch nicht genannte Aktivitäten:</i> -
	E2	VBN: Definieren von Zielen einer benutzerorientierten Gestaltung FBN: Abgleichen von Nutzungsanforderungen zu Unternehmensanforderungen (bspw. Datenschutz, IT-Sicherheit, Performanz) BGL: Festlegen des Evaluationsgegenstandes und überprüfen, ob dieser zuverlässig messbar ist BGL: Definieren der Zielsetzung einer Evaluation
F.13		<i>siehe Anhang F.2 - Tabelle F.3</i>
F.14	E1	VBN: Nutzer FBN: Beschriebener Nutzungskontext EGL: Nutzungsanforderungen BGL: Prototypen
	E2	VBN: Zeitliche Planung, Integration mit anderen Abteilungen FBN: Beschriebener Nutzungskontext EGL: Anforderungen BGL: Gestaltungslösungen
F.15	E1	VBN: Beschreibung des Nutzungskontextes FBN: Anforderungsspezifikation EGL: Beschreibung der Interaktion, der Benutzerschnittstelle BGL: Evaluationsergebnisse, Ergebnisse der Konformitätsprüfung
	E2	VBN: Kontextbeschreibung FBN: Anforderungsspezifikation EGL: Informationsarchitektur, Interaktionsspezifikation, Spezifikation der Benutzerschnittstelle BGL: Ergebnisse der Nutzertests, Ergebnisse der Konformitätsprüfung
F.16	E1	<i>Ergänzende, im Modell noch nicht genannte Artefakte:</i> -
	E2	EGL: Technische Rahmenbedingungen

Weiter auf der nächsten Seite

Frage	Experte	Aussage
F.17		<i>siehe Anhang F.2 - Tabelle F.4</i>
F.18	E1	Ja! Man muss auch Kontextdaten, Erfordernisse und Anforderungen „evaluieren“, besser „validieren“, im Sinne von „werden die Daten durch die Nutzergruppe bestätigt?“
	E2	Jedes Zwischenergebnis (z.B Kontext oder Anforderungen) sollte mit Usern VALIDIERT werden, ich würde EVALUATION aber eher auf das spezifische Lösungsdesign (auch prototypisch) beschränken.
F.19	E1	Absolut. Alles andere ist wie „einen Nutzer durch einen Raum rennen lassen und gucken, wo er überall gegen rennt“. Wenn man das dann behoben hat, weiss man immer noch nicht, was der Mensch in dem Raum hätte tun können müssen.
	E2	Unbedingt, viele Evaluationen machen nur Sinn, wenn man sie auf den Nutzungskontext und konkrete Nutzungsanforderungen bezieht. Usability als Qualität wird ja ebenfalls nur auf spezifische Aufgaben in definiertem Nutzungskontext bezogen (in der 9241).
F.20	E1	Es ist eine „Inspektion gegen Konventionen“, das ist legitim. So kann man formale Designmängel identifizieren, die ohnehin Probleme machen. Ein guter erster Schritt.
	E2	Oberflächlich, kontextfrei, „Pixel-Ergonomie“
F.21	E1	Sehr wichtig, wenn dann die Nutzungskontextdaten auf enthaltene Erfordernisse hin untersucht werden, die dann wiederum zur Spezifikation von Nutzungsanforderungen verwendet werden, (vgl. Leitfaden Usability der DAkkS). „Analyse“ des Nutzungskontextes heißt „Analysieren, welche Erfordernisse im Nutzungskontext stecken“. Dazu muss vorher eine „Beschreibung des Nutzungskontextes“ vorliegen, damit diese überhaupt analysiert werden kann. Beste Beschreibungsform: Kontextszenario (vgl. Leitfaden Usability der DAkkS).
	E2	Sie liefert viele Perspektiven/Dimensionen, die in der klassischen Software-Entwicklung nicht betrachtet werden, und können somit helfen, Probleme zu lösen, die sonst nicht oder nicht gut gelöst werden könnten.
F.22	E1	Eine Lösung ist nur so gut, wie sie ein Problem löst. Wird das Problem nicht verstanden, kann es nicht gelöst werden. Das beste Verständnis eines Problems ist das zu befriedigende Erfordernis.
	E2	Ohne Kenntnis des Problems ist eine gute Lösung eher Zufall. Je größer die Kenntnis ist, desto systematischer kann man Lösungen entwickeln und qualitätssichern. Heisst: Per Zufall kann es zu sehr guten Lösungen kommen, aber mit großem Risiko.
F.23	E1	Nein, hängt damit zusammen, dass die Lifecycle-Modelle (vgl. ISO/IEC 15288) immer früher anfangen (z.B. beim organisatorischen Bedarf für IT-Unterstützung) und viel später aufhören (z.B. bei der „Beerdigung des Systems“).
	E2	Nein, ich halte sie nicht einmal für das UE geeignet, Phasen abzubilden. Ich halte alle vier für Aktivitäten, die in mehr oder weniger großer Ausprägung über den gesamten Entwicklungszeitraum passieren.

Weiter auf der nächsten Seite

Frage	Experte	Aussage
F.24	E1	Im Grundsatz ja. Bloss, dass die Aktivitäten im SE nicht aus der Perspektive der Nutzung durchgeführt werden, sondern aus der Perspektive der technischen Umsetzung.
	E2	Nur zu 20-30 Prozent, den Rest halte ich für spezifisch.
F.25	E1	UE-Outputs sind typisch SE-Inputs. Der Informationsaustausch zwischen UE/SE steigert die Validität (Richtigkeit) des zu entwickelnden Systems. Ohne diesen Informationsaustausch wird lediglich sichergestellt dass „richtig entwickelt wird“ (Verifikation).
	E2	Ja, ich halte es sogar für zwingend notwendig, Synchronisationspunkte zu haben, und nicht nur Informationen, sondern auch Arbeitsprodukte auszutauschen - das würde sowohl SE->UE, wie auch UE->SE heissen.
F.26	E1	Den RE-Prozess durch Nutzungskontext und Nutzungsanforderungen komplettieren. Sämtliche Interaktionsspezifikationen „unter der Haube“ auf der Basis durchspezifizierter Interaktionsszenarien „über der Haube“ (= Nutzungsszenarien) erstellen.
	E2	Kontext <-> technische Rahmenbedingungen (z.B. Architektur, Service Management, etc.), Nutzungsanforderungen <-> Business Anforderungen, Nicht funktionale Anforderungen (z.B. Datenschutz, Performance, etc.), Design Solution <-> technische Lösungsspezifikation, Usability Evaluation <-> System-/Integrationstest
F.27	E1	Ja, absolut. Die meisten Anforderungsspezifikationen unterscheiden nicht systematisch zwischen „Stakeholder-Anforderungen“ und „Systemanforderungen“. Innerhalb der Klasse „Stakeholder-Anforderungen“ werden die Nutzungsanforderungen typisch völlig ignoriert.
	E2	Durch ein gemeinsames Dokument wird das Commitment der Organisation, die Usability Anforderungen umzusetzen, erhöht. Ebenso werden Widersprüche oder Inkonsistenzen vermieden, die sonst Mehraufwand oder Interpretationsspielräume liefern würden. Es sollte allerdings idealerweise auch für die anderen Produkte des UCD (u.a. die UI Spezifikation) eine Integration mit den im SE verwendeten Dokumenten geben, da die Lösungsumsetzung meist nicht (nur) auf der Anforderungsspezifikation basiert, sondern eher auf der Lösungsspezifikation.

Tabelle F.5.: Aussagen der Experten

G. Modelle

Abbildungen der Teilmodelle des in dieser Arbeit entwickelten ganzheitlichen Integrationsmodells.

Erläuterungen zu Abbildungen G.1 bis G.20:

- Graue Balken: Phasen
- Quadrate: Aktivitäten
- Fünfecke: Artefakte
- Grün: Aktivitäten und Artefakte des Usability Engineering
- Blau: Aktivitäten und Artefakte des Software Engineering
- Magenta: Aktivitäten und Artefakte des Marketing und der Marktforschung
- Braun: Aktivitäten und Artefakte des Usability Engineering, die auf Basis des Abgleiches mit den Konformitäts- und Rahmenanforderungen nach Nebe (2009) und den Expertenbefragungen erstellt wurden
- Abbildungen G.1 bis G.5: Teilmodell der Aktivitäten
- Abbildungen G.6 bis G.10: Teilmodell der Aktivitäten und Artefakte aus Perspektive des UE
- Abbildungen G.11 bis G.15: Teilmodell der Aktivitäten und Artefakte aus Perspektive des SE
- Abbildungen G.16 bis G.20: Teilmodell der Artefakte und deren Lebenszyklus

Erläuterungen zu Abbildung G.21:

- Modell zur Verdeutlichung der Relevanz des UE in sämtlichen Phasen der Entwicklung
- weiß: erzeugte Modelle im UE
- grau: existierende Modelle, die im UE angewandt oder berücksichtigt werden
- Wolken: Evaluationstechniken



Abbildung G.1.: Teilmodell von Aktivitäten des UE (grün, braun) & SE (blau) mit potentiellen Aktivitäten des Marketing & der Marktforschung (magenta) - Übersicht



Abbildung G.2.: Teilmodell von Aktivitäten des UE & SE - Teil I

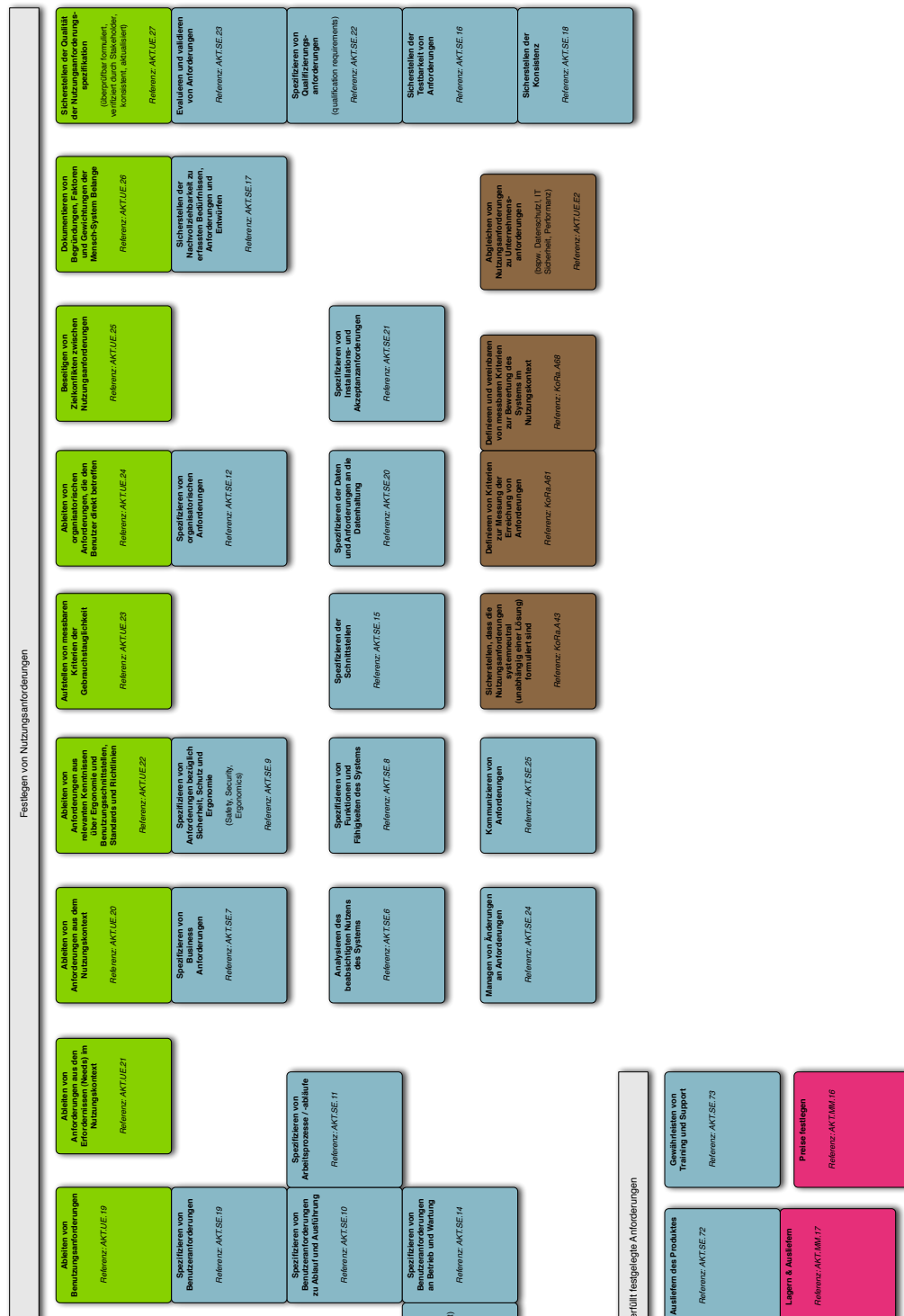


Abbildung G.3.: Teilmodell von Aktivitäten des UE & SE - Teil II



Abbildung G.4.: Teilmodell von Aktivitäten des UE & SE - Teil III

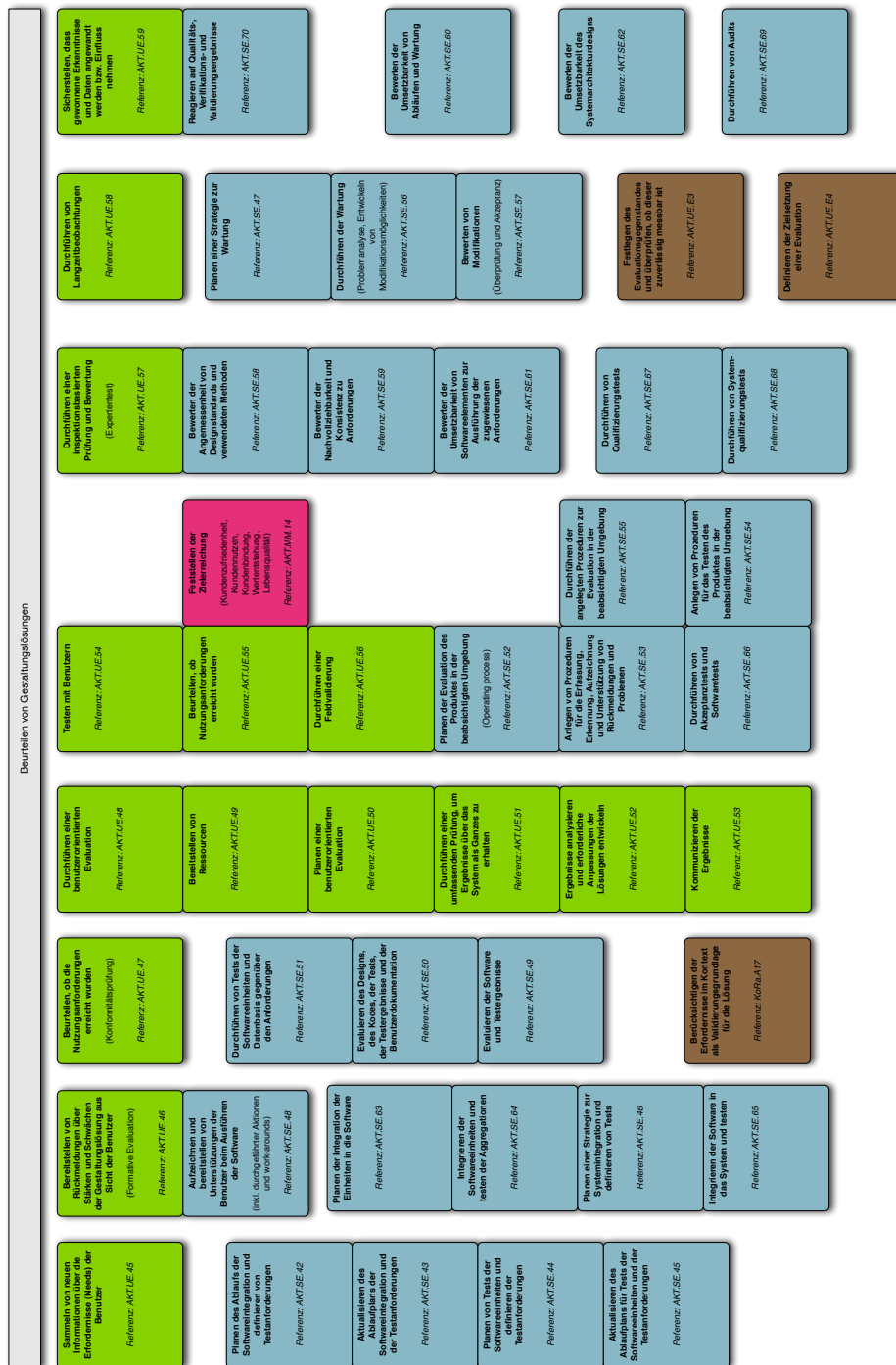


Abbildung G.5.: Teilmodell von Aktivitäten des UE & SE - Teil IV

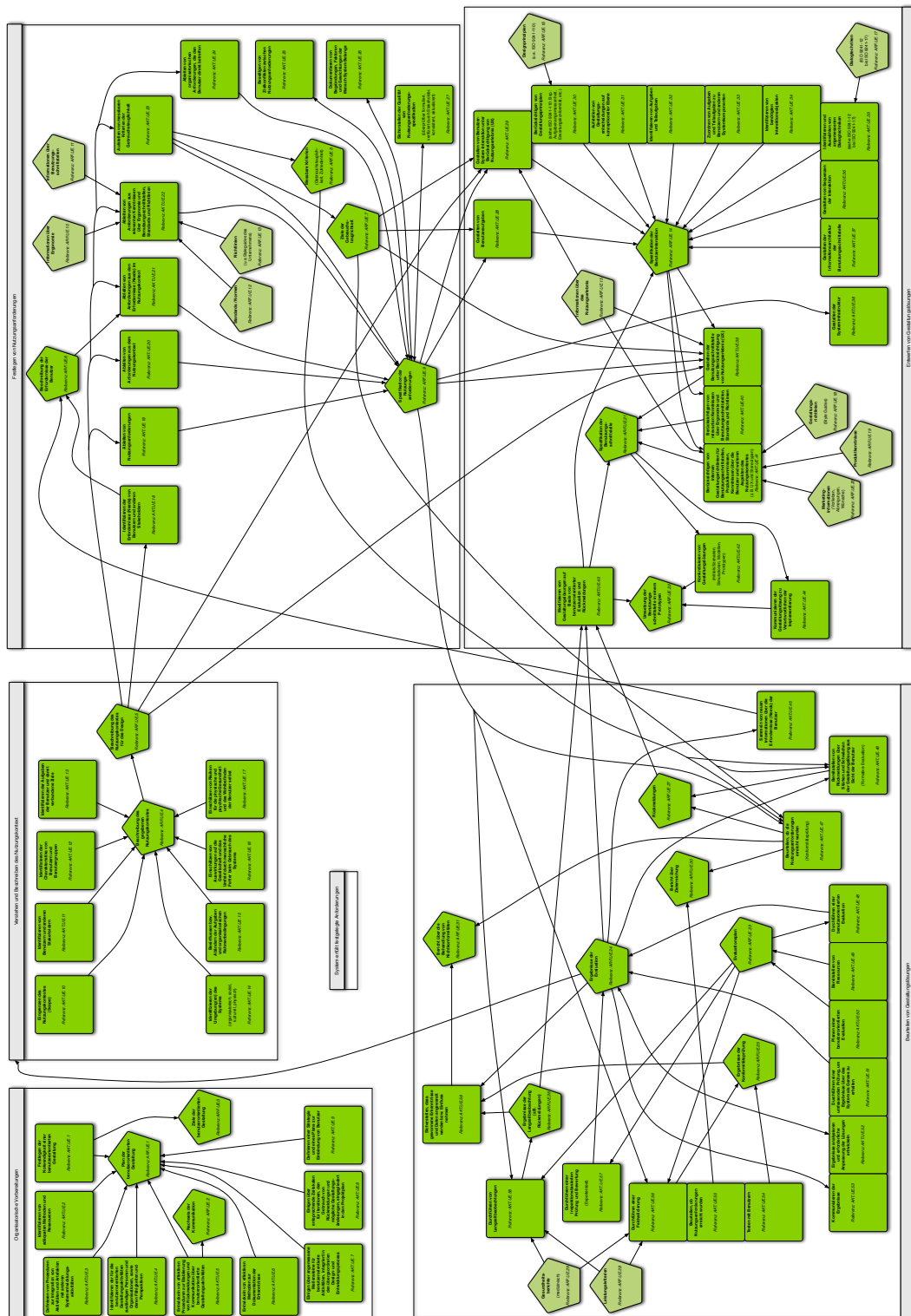


Abbildung G.6.: Teilmodell von Aktivitäten und Artefakte des UE - Übersicht

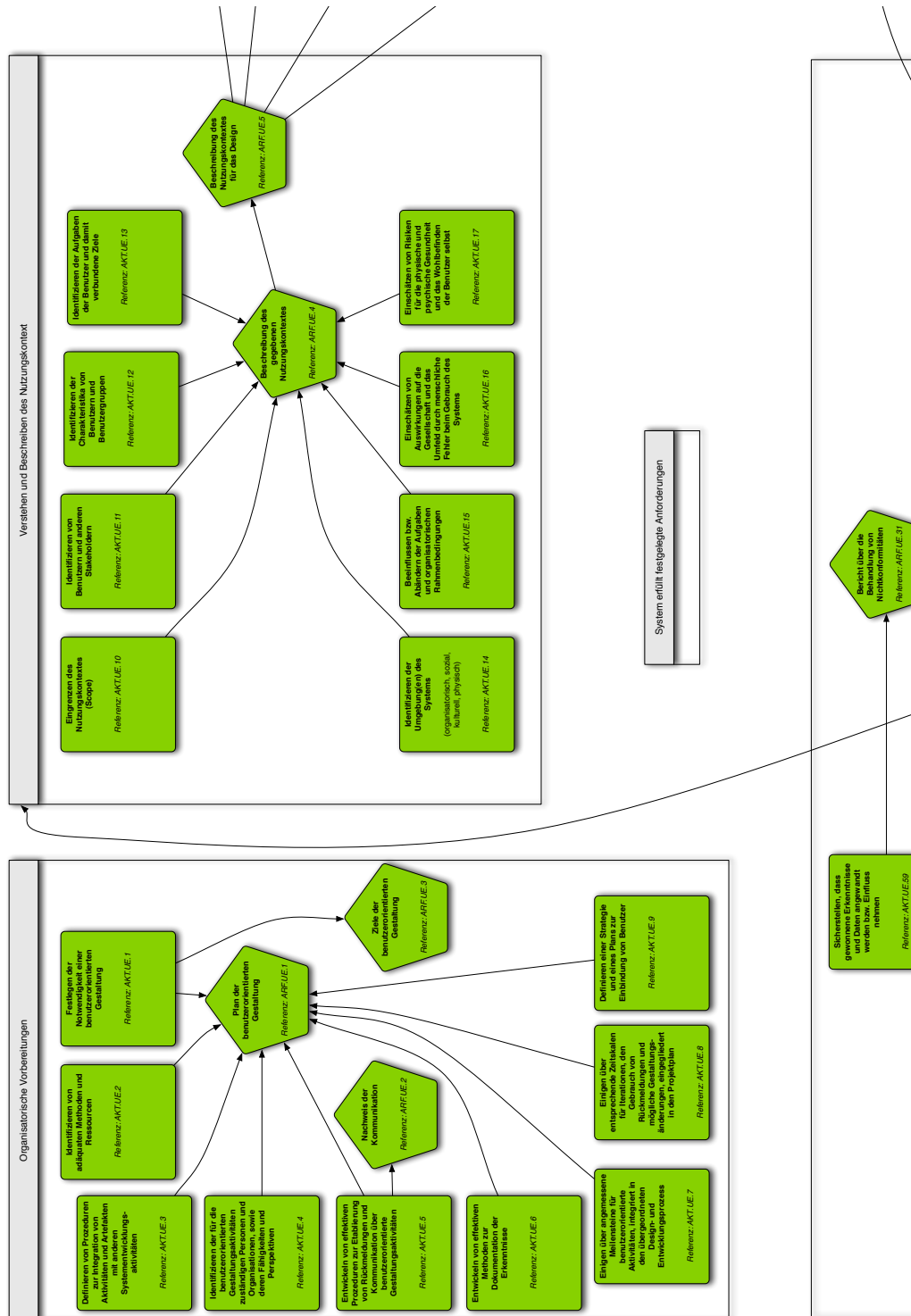


Abbildung G.7.: Teilmodell von Aktivitäten und Artefakten des UE - Teil I

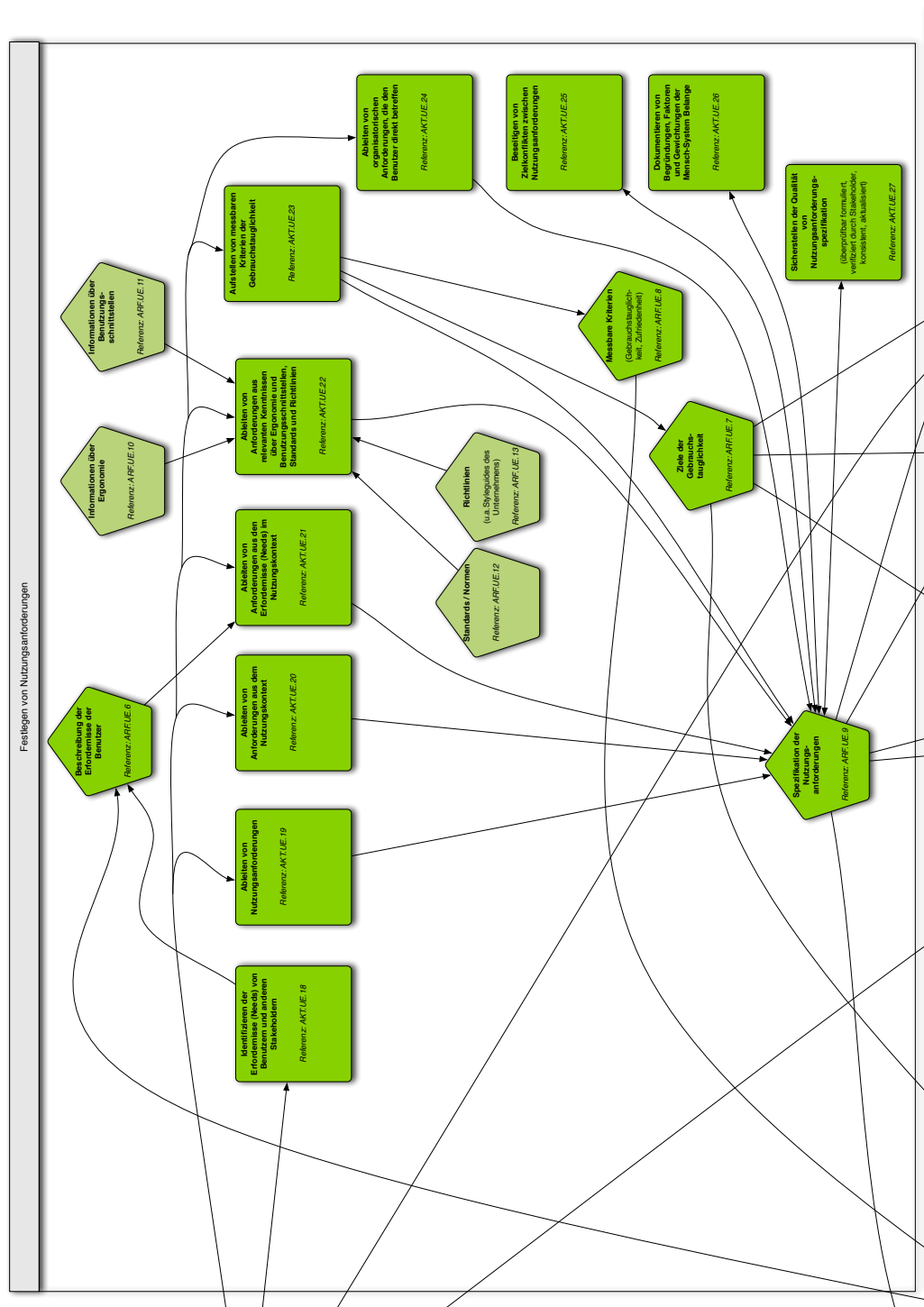


Abbildung G.8.: Teilmodell von Aktivitäten und Artefakte des UE - Teil II

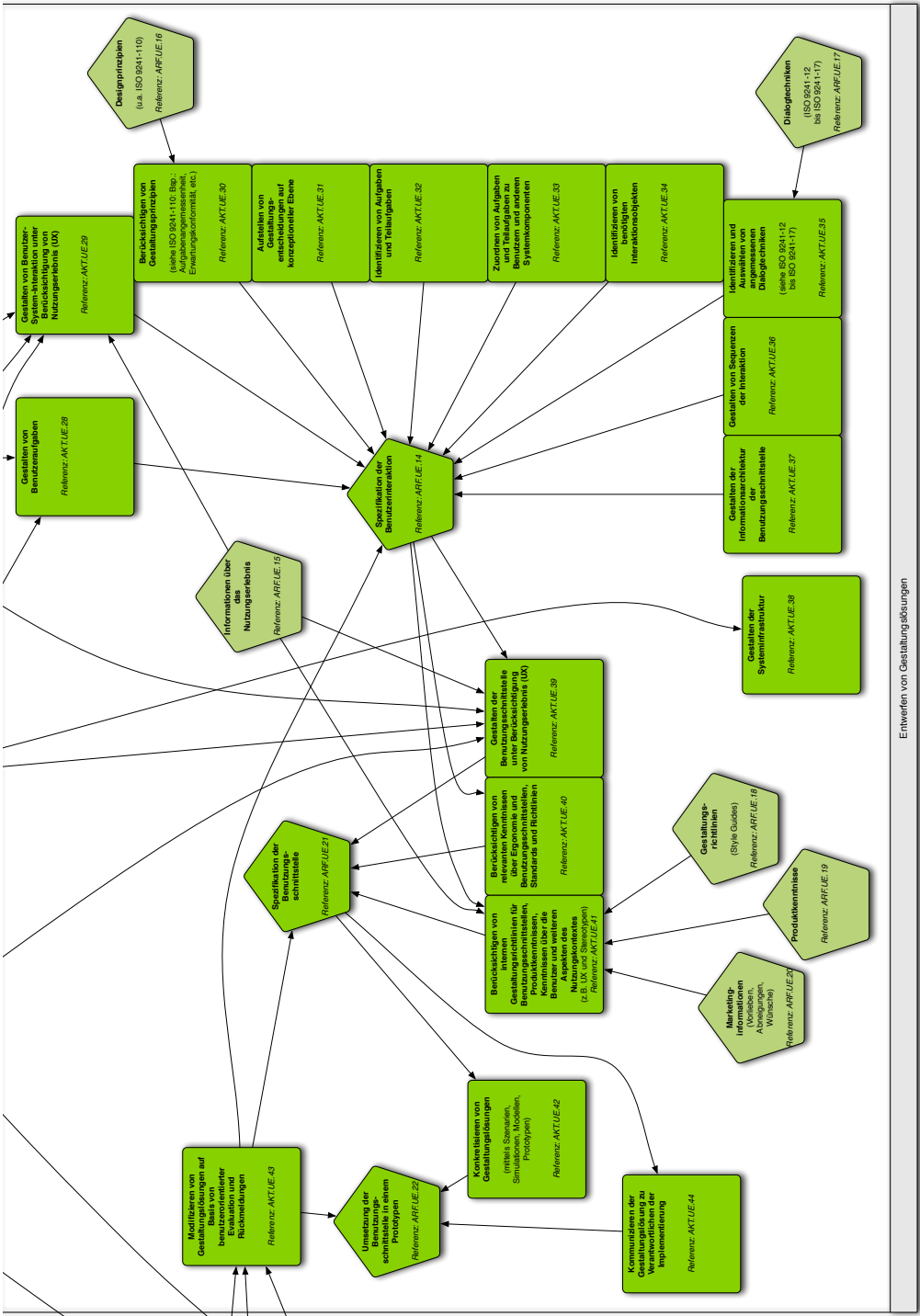


Abbildung G.9.: Teilmodell von Aktivitäten und Artefakte des UE - Teil III

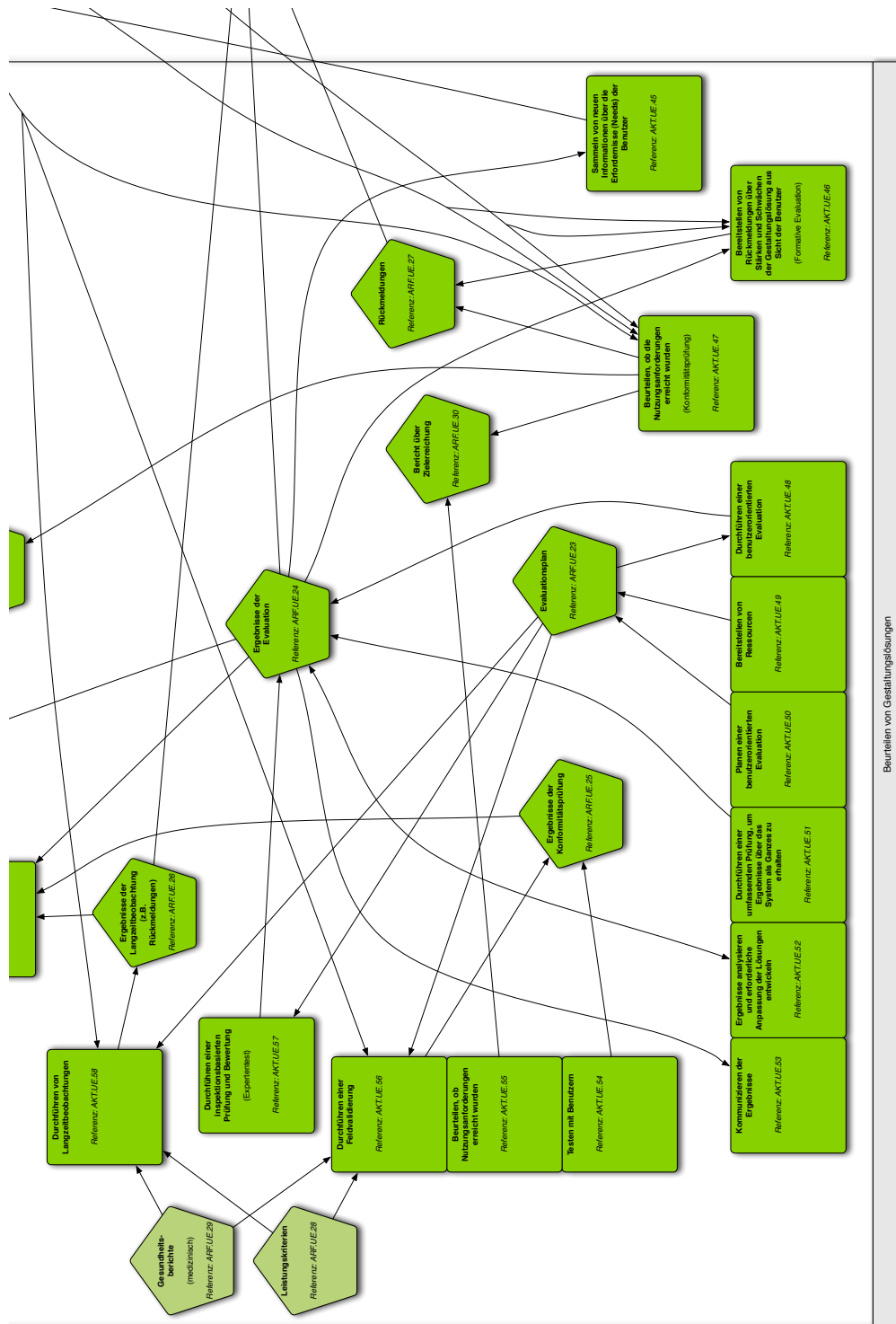


Abbildung G.10.: Teilmodell von Aktivitäten und Artefakte des UE - Teil IV

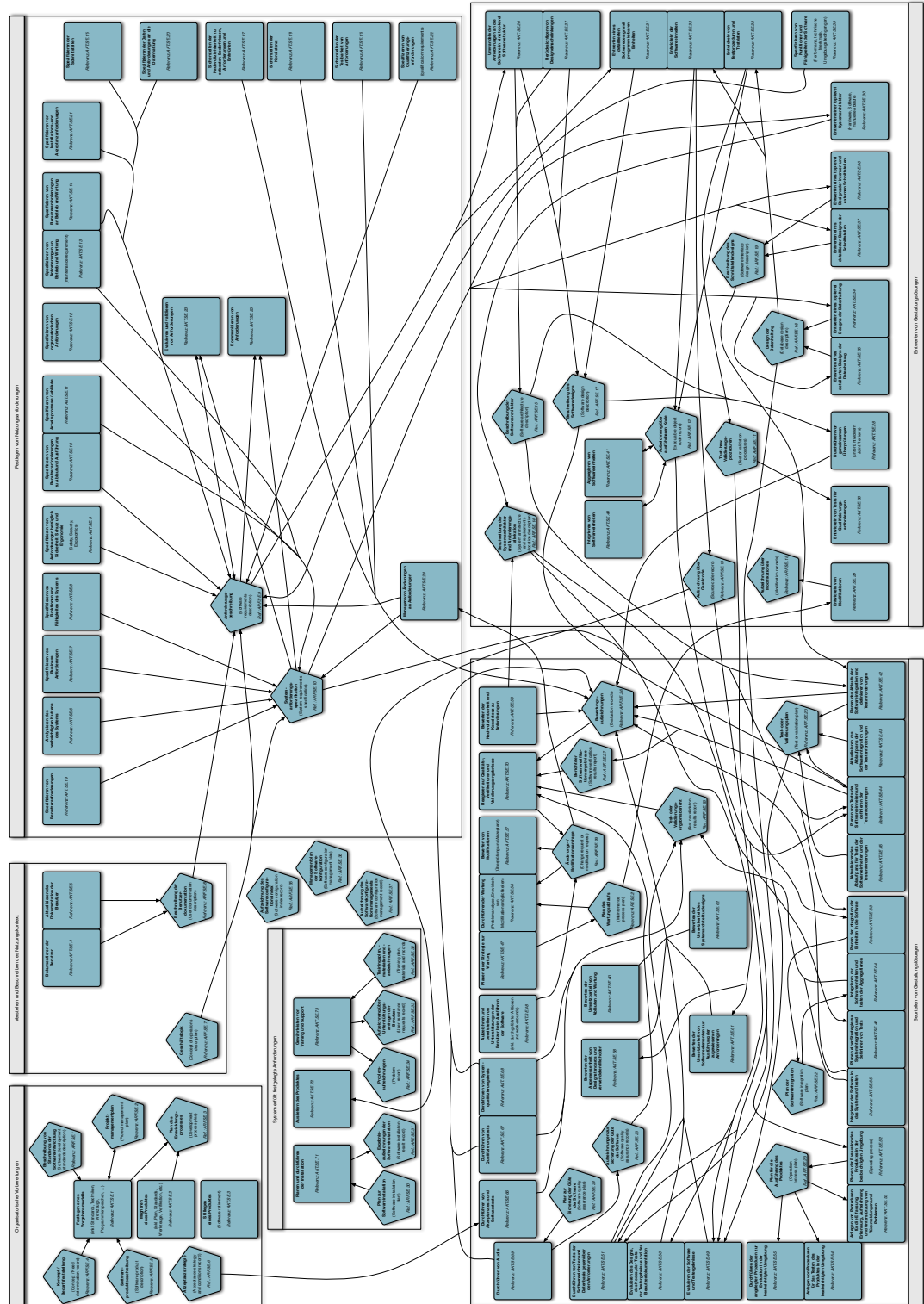


Abbildung G.11.: Teilmodell von Aktivitäten und Artefakte des SE - Übersicht

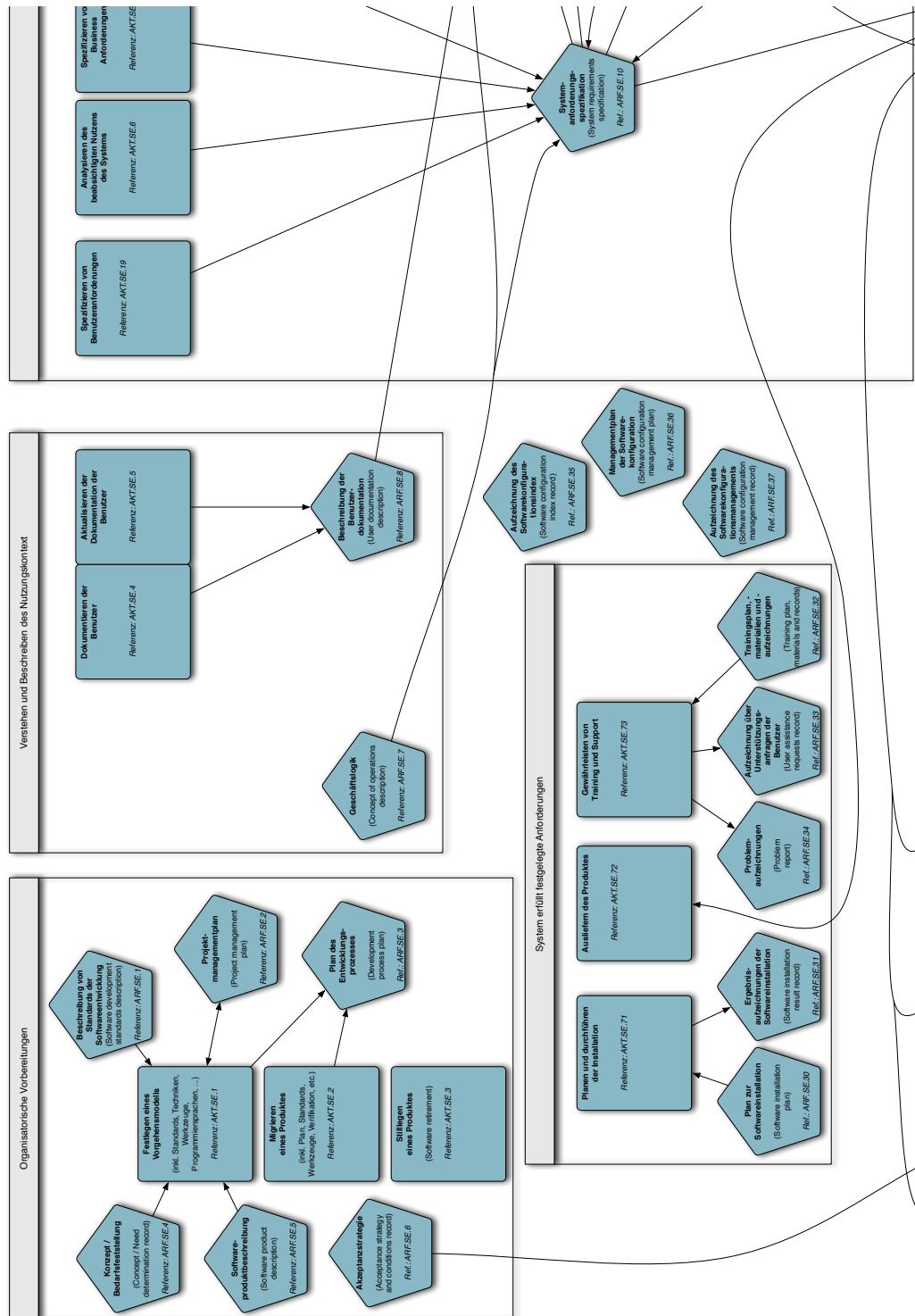


Abbildung G.12.: Teilmodell von Aktivitäten und Artefakte des SE - Teil I

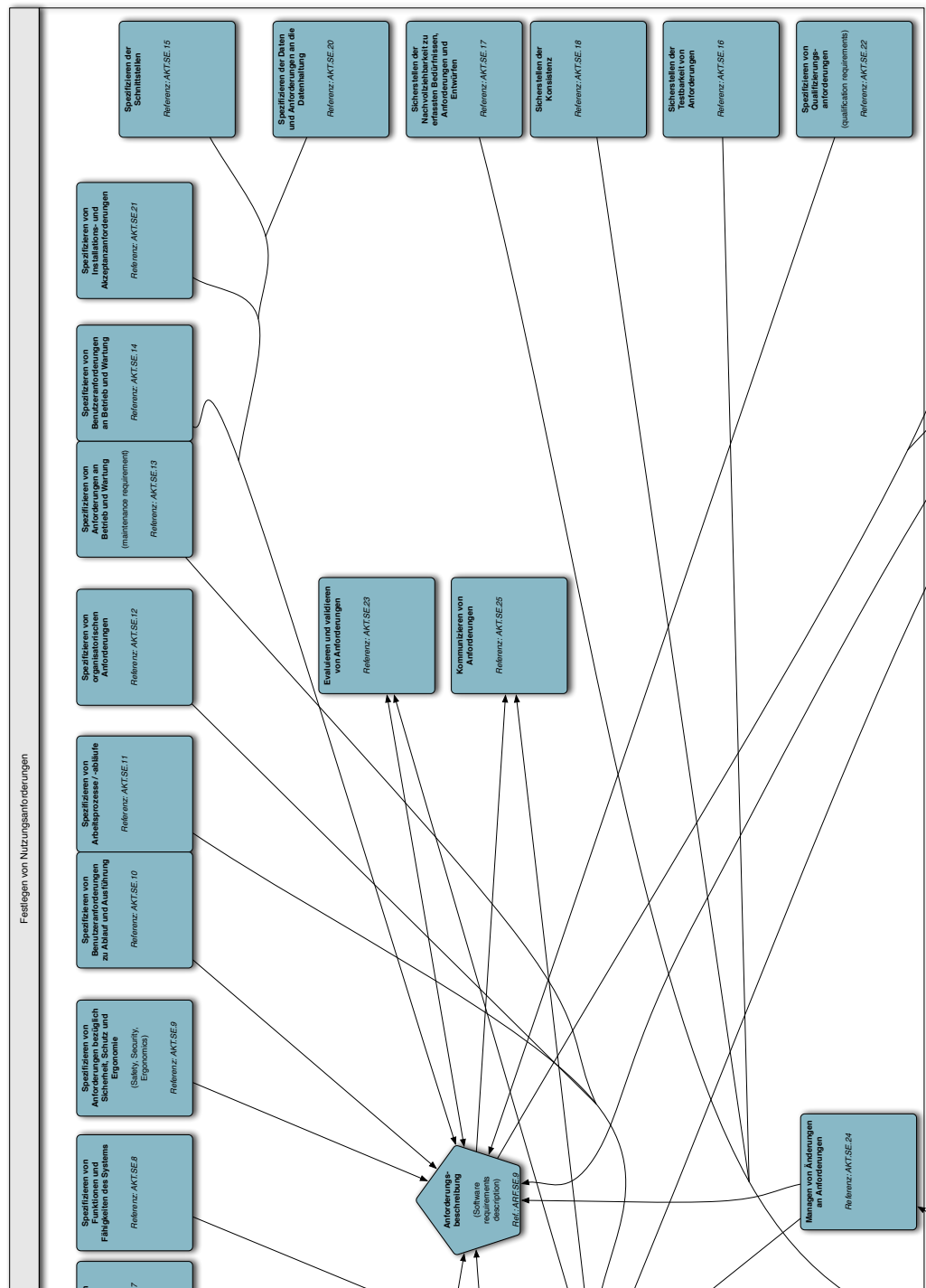


Abbildung G.13.: Teilmodell von Aktivitäten und Artefakte des SE - Teil II

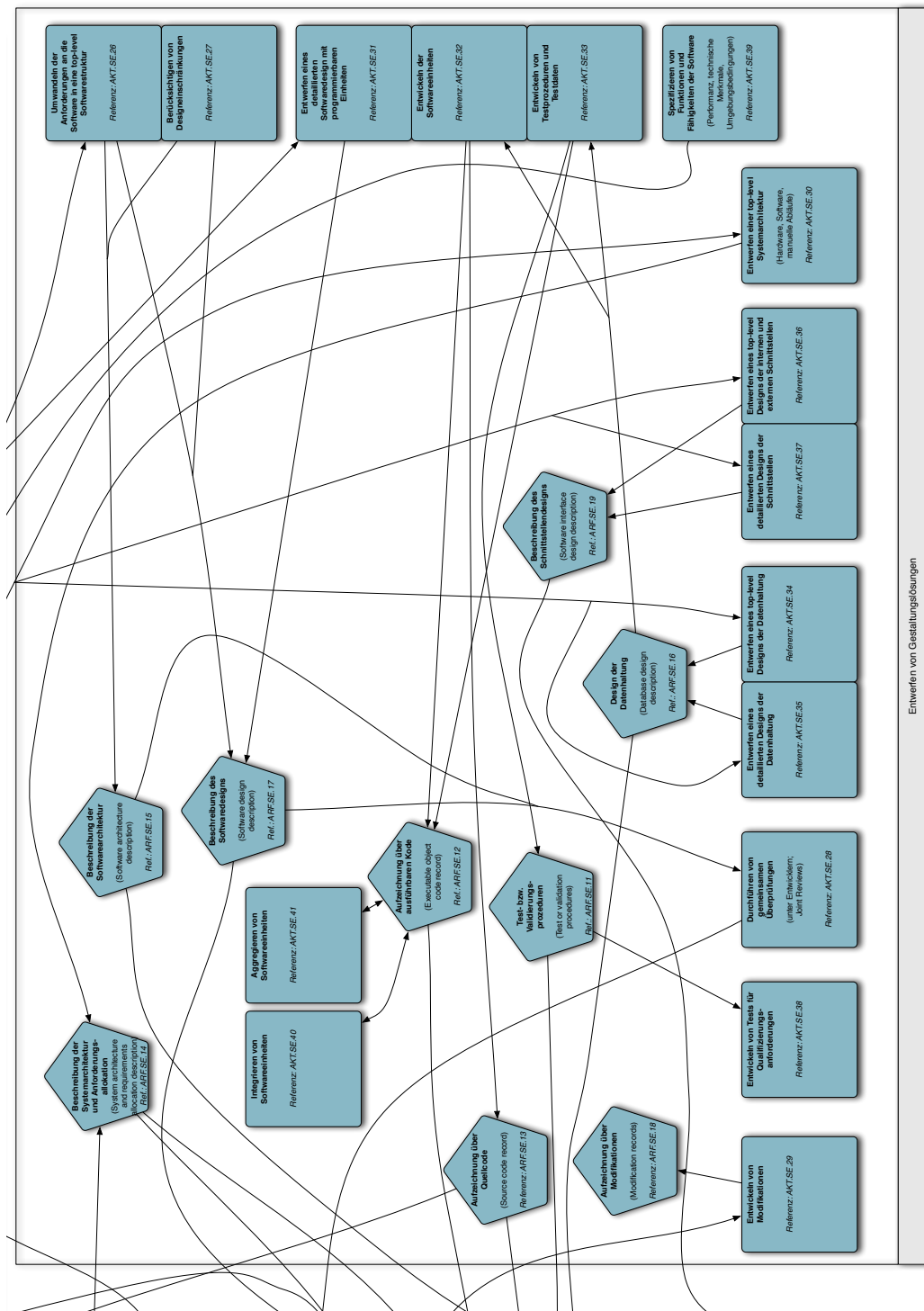


Abbildung G.14.: Teilmodell von Aktivitäten und Artefakte des SE - Teil III

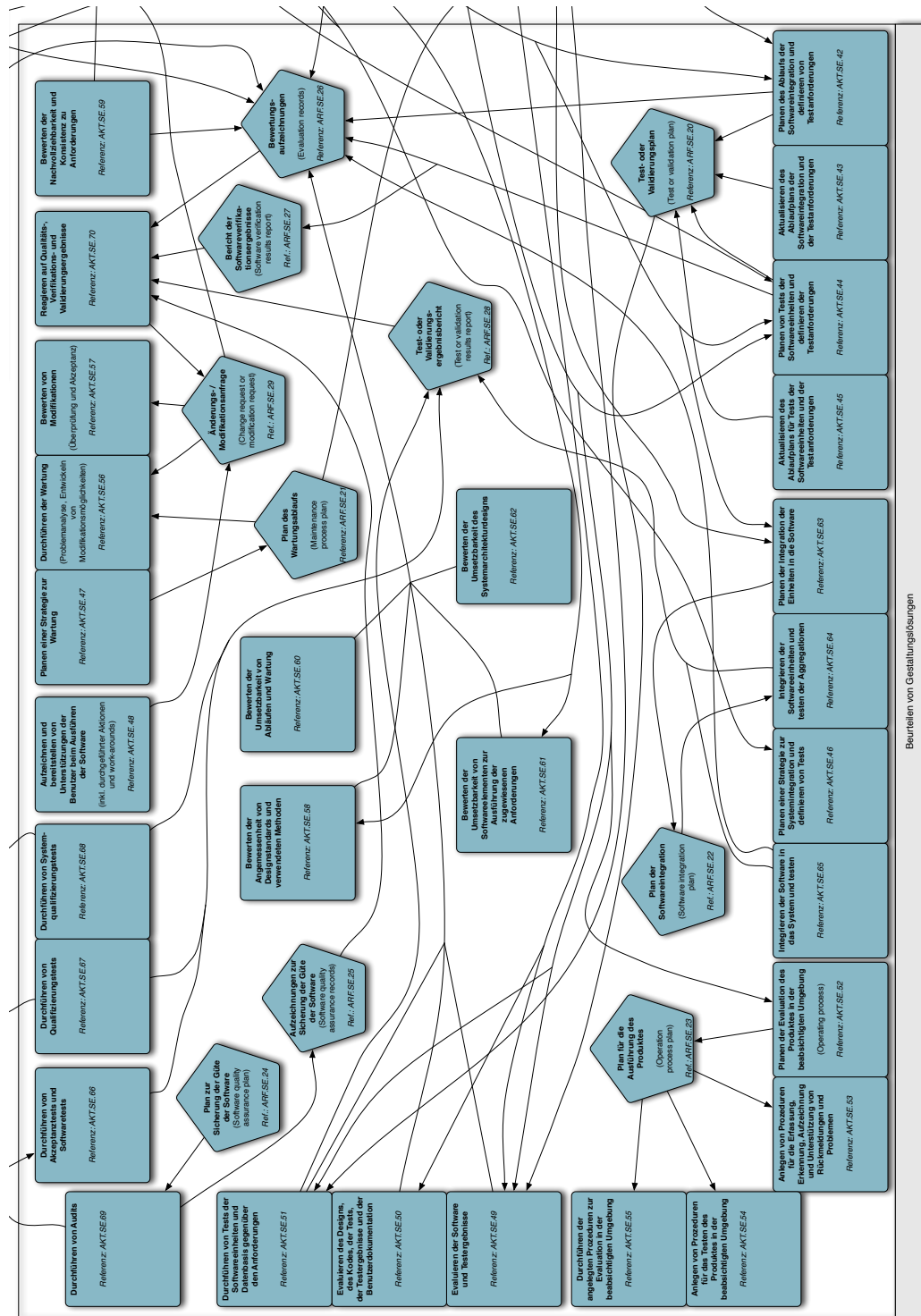


Abbildung G.15.: Teilmodell von Aktivitäten und Artefakte des SE - Teil IV

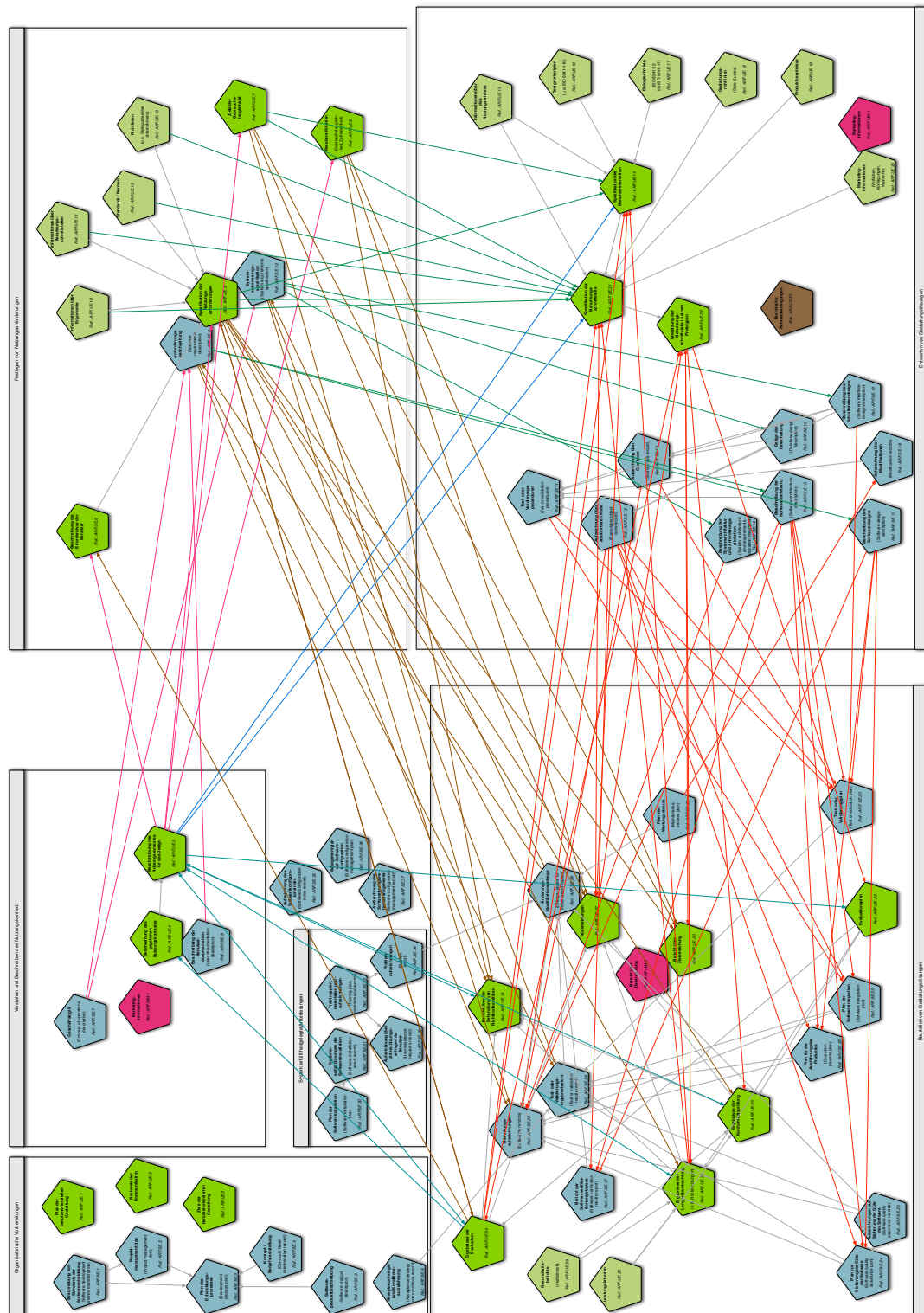


Abbildung G.16.: Teilmodell von Artefakten des UE (grün, braun) & SE (blau) mit potentiellen Aktivitäten des Marketing & der Marktforschung (magenta) - Übersicht

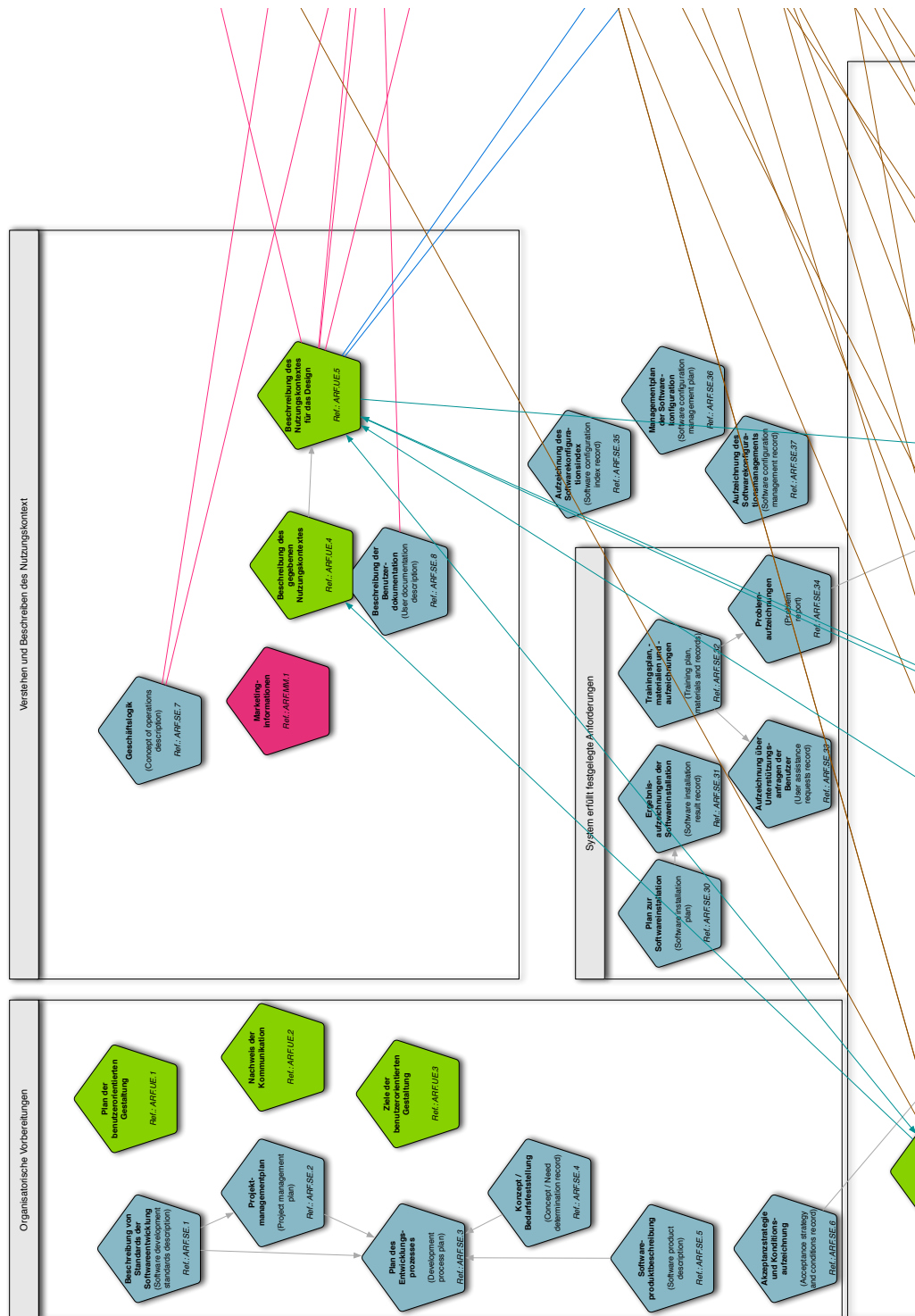


Abbildung G.17.: Teilmodell von Artefakten des UE & SE - Teil I

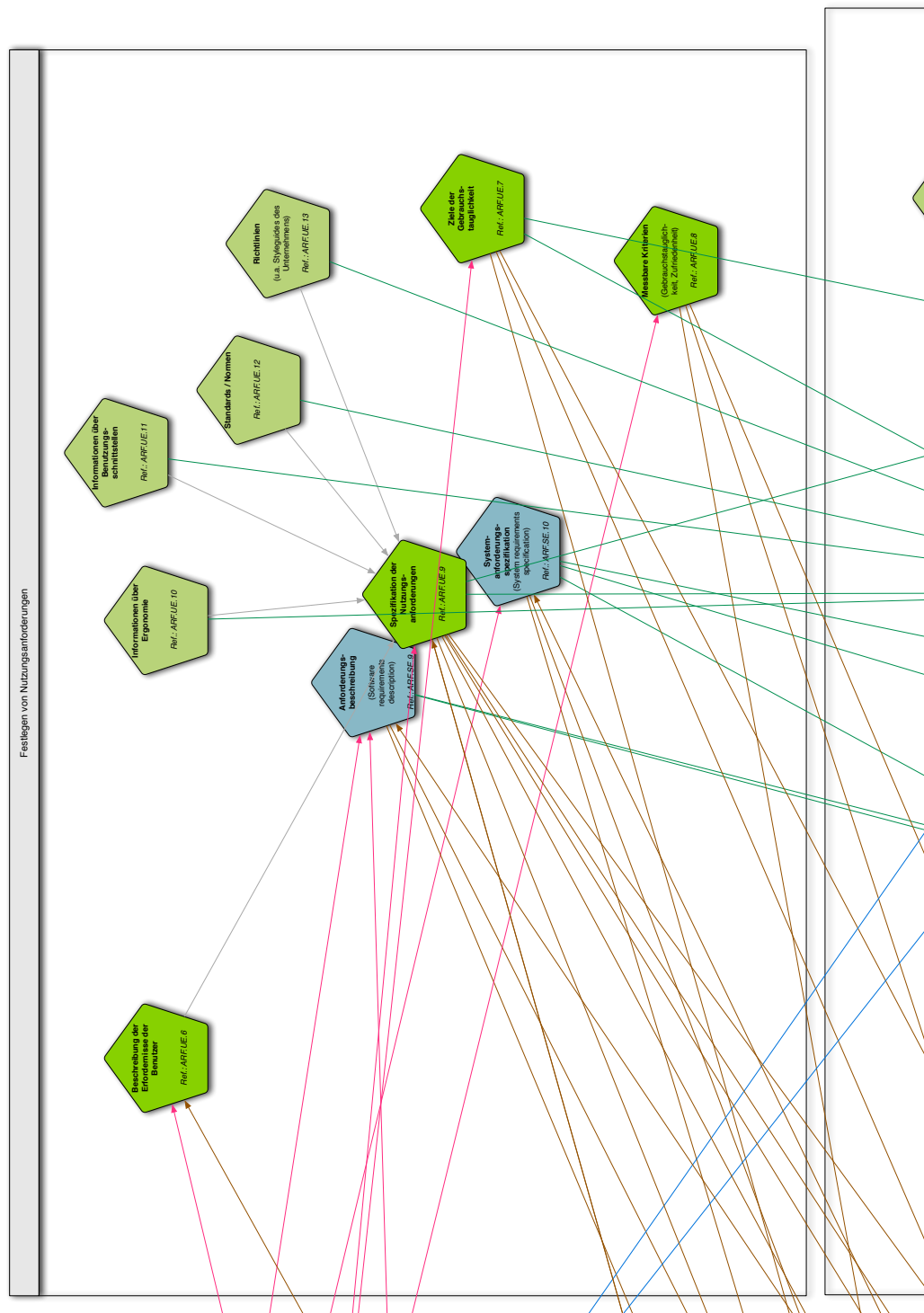


Abbildung G.18.: Teilmodell von Artefakten des UE & SE - Teil II

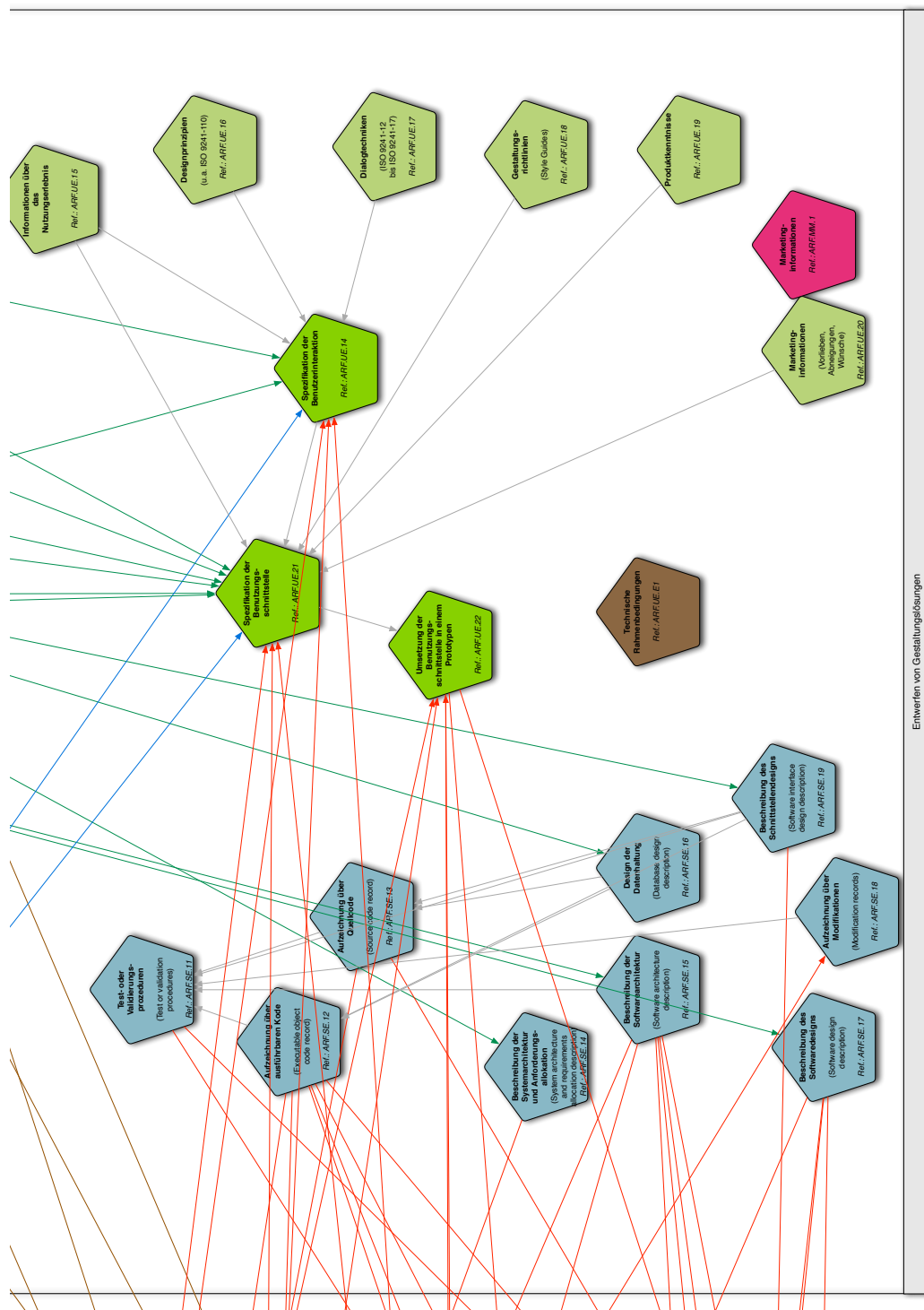
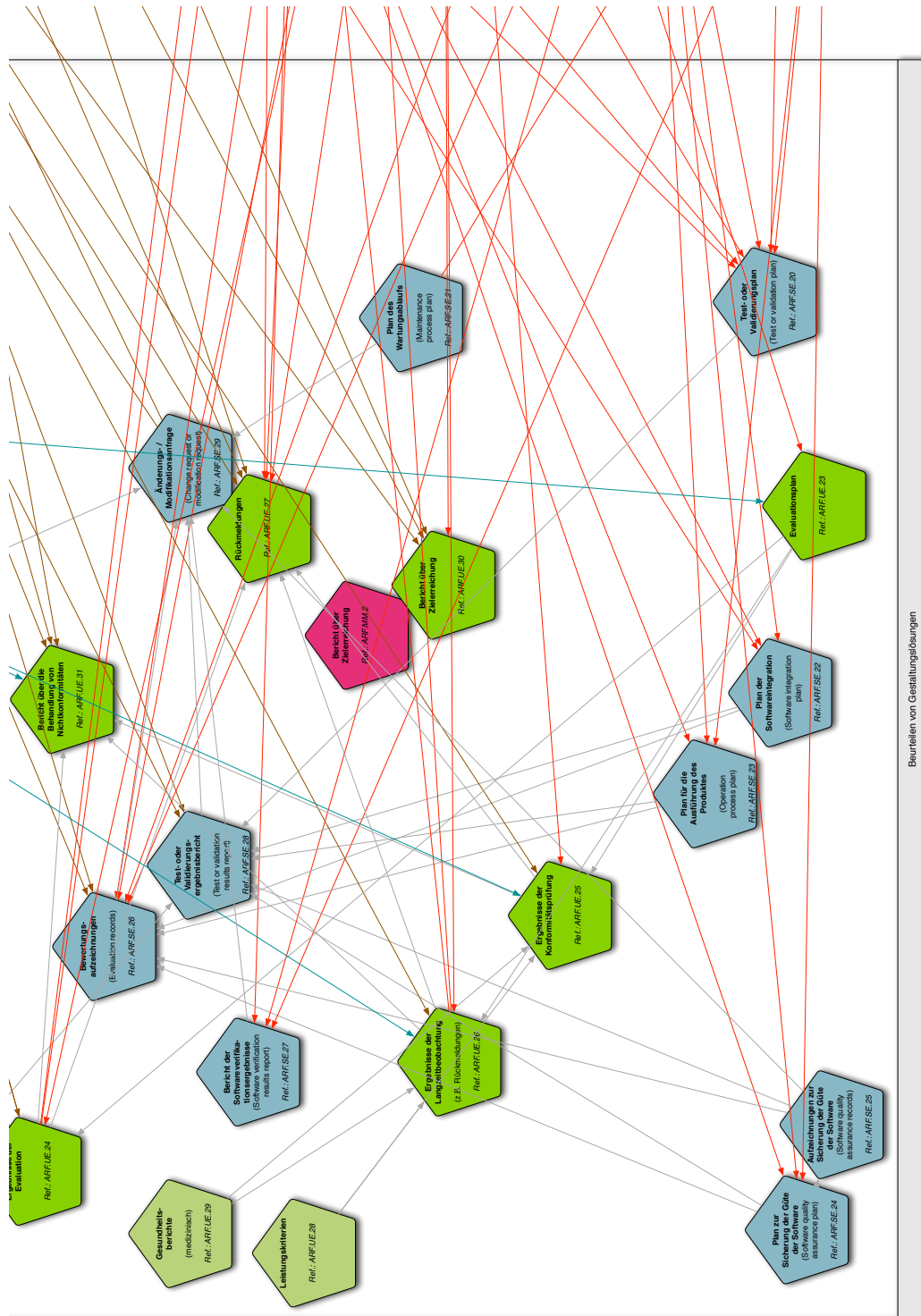


Abbildung G.19.: Teilmodell von Artefakten des UE & SE - Teil III



Beurteilen von Gestaltungsdesignen

Abbildung G.20.: Teilmodell von Artefakten des UE & SE - Teil IV

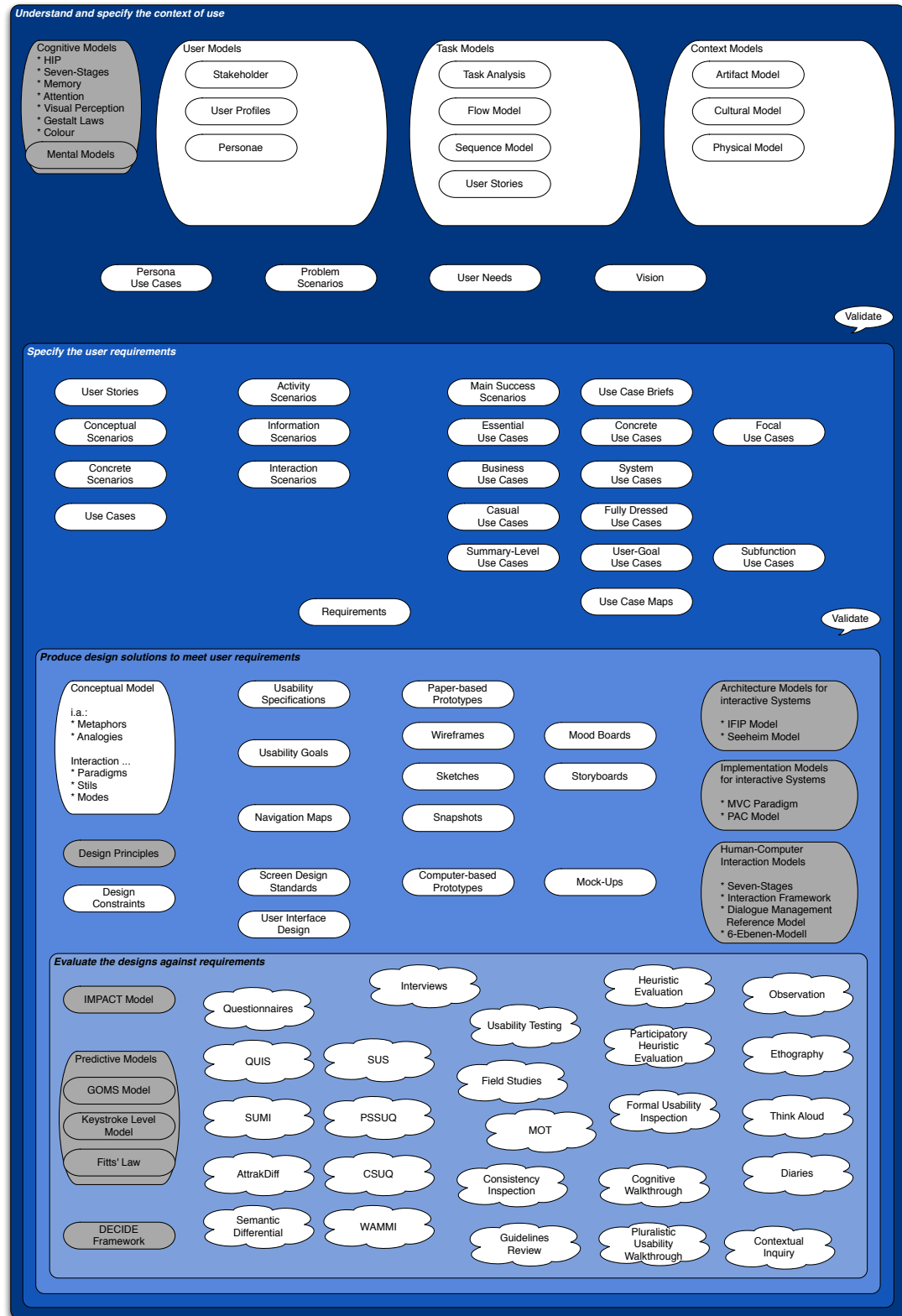


Abbildung G.21.: Überblick über die Modelle des UE und deren phasenbedingten Abhängigkeiten (weiß = erzeugte Modelle; grau = existierende Modell, die Einfluss nehmen; Wolken = Evaluationstechniken)

H. Glossar

Dieses Glossar bezieht sich auf der einen Seite auf Abbildung G.21 in Anhang G und definiert die darin enthaltenen Modelle und Evaluationstechniken. Des Weiteren enthält das Glossar aber auch Definitionen in Bezug zur semantischen Analyse als Vorarbeit zur Expertenbefragung (Kapitel F). Die Definitionen wurden zum großen Teil aus ihren Originalquellen zitiert, wodurch die Sprache zwischen deutsch und englisch abwechselt.

6-Ebenen-Modell	Beschreibt das mentale Modell eines mit einem Computersystem handelnden Menschen hinsichtlich Planung und Durchführung von Handlungen und Wahrnehmung von Systemzuständen und ihren Änderungen. (Herczeg 2009)
Activity Scenario	Narratives of typical or critical services that people will seek from the system. They deliberately focus on pure functionality, refraining from specifying details about what the system will look like or how users will manipulate it. (vgl. Rosson & Carroll 2002)
Aktivität	Eine Aktivität bezeichnet eine in sich geschlossene Abfolge von (Teil-)Tätigkeiten, die von wenigen Personen in einer gewissen Zeit durchführbar sind und ein festgelegtes Ergebnis liefern. (Winter 2005)
Artefakt	Ein Artefakt bezeichnet ein Arbeitsergebnis, welches durch Ausübung einer Aktivität entsteht. Artefakte sind dynamischer Natur und können sich im Laufe eines Projektes verändern, wachsen oder phasenübergreifend weiter entwickelt werden. (Winter 2005)
Artifact Model	Consolidated artifact models shows common organizing themes and concepts that people use to pattern their work. (Beyer & Holtzblatt 1997)
AttrakDiff	Das Attrakdiff2 gehört zur Klasse der semantischen Differenziale. Es besteht aus 28 bipolaren, siebenstufigen Items. Diese 28 einzelnen Items werden zu vier Skalen mit jeweils sieben Items zusammengefasst. Sie operationalisieren die folgenden Konstrukte: Pragmatische Qualität, Hedonische Qualität - Stimulation, Hedonische Qualität - Identität, Attraktivität. (Hassenzahl et al. 2008)

Aufgabe	Benötigte Aktivitäten zur Erreichung eines Ziels. (ISO 9241-210:2010)
Benutzer	Die Person, die mit dem Produkt arbeitet. (ISO 9241-11:1998)
Benutzungsschnittstelle ...	Alle Bestandteile eines interaktiven Systems (Software oder Hardware), die Informationen und Steuerelemente zur Verfügung stellen, die für den Benutzer notwendig sind, um eine bestimmte Arbeitsaufgabe mit dem interaktiven System zu erledigen. (ISO 9241-210:2010)
Business Use Case	The phrase business use case is a shortcut indicating that the use case emphasizes the operation of the business rather than the operation of a computer system. It is possible to write a business use case at any goal level, but it can only be at enterprise or organization scope. (Cockburn 2000)
Casual Use Case	A casual use case it written as a simple, prose paragraph. It is likely to be missing project information associated with the use case, and it is likely to be less rigorous in its description than is a fully dressed use case. (Cockburn 2000)
Cognitive Model	A representation of some aspect of the mind, involving the acquisition of knowledge (e.g. understanding, remembering, reasoning, learning). (Preece et al. 1994)
Cognitive Walkthrough ...	Cognitive walkthroughs involve simulating a user's problem-solving process at each step in the human-computer dialog, checking to see if the user's goals and memory for actions can be assumed to lead to the next correct action. (Preece et al. 2007)
Computer-based Prototype	Computer-based prototyping provides a version of the system with limited functionality so that users can actually interact with it. (Preece et al. 1994)
Conceptual Model	Conceptual models is the generic term that describes the various ways in which systems are understood by different people. Primarily these consist of (i) the way users conceptualize and understand the system and (ii) the way designers conceptualize and view the system. (Preece et al. 1994)
Conceptual Scenario	Conceptual Scenarios are a combination of similar user stories, achieve an higher level of abstraction and are useful for generating design ideas while understanding the requirements of the system. (vgl. Benyon et al. 2005)

Concrete Scenario	Concrete Scenarios are an elaboration of one conceptual scenario and indentify most eventualities while allocating functions between people and devices. (vgl. Benyon et al. 2005)
Concrete Use Case	Detailed description of a task, but (unlike task scenarios) they are not personalized and describe the use of a system at a more generic level. (Constantine & Lockwood 1999)
Consistency Inspection . . .	The experts verify consistency across a family of interfaces, checking the terminology, fonts, color schemes, layout, input and output formats, and so on within the interfaces as well as in the documentation and online help. (Shneiderman & Plaisant 2009)
Contextual Inquiry	Contextual Inquiry is based on a set of principles that allow it to be molded to each situation that a project encounters: <i>context</i> , go to the customers' workplace and watch them do their own work; <i>partnership</i> , talk to them about their work and engage them in uncovering unarticulated aspects of work; <i>interpretation</i> , develop a shared understanding with the customer about the aspects of work that matter; and <i>focus</i> , direct the inquiry from a clear understanding of your own purpose. (Beyer & Holtzblatt 1997)
CSUQ	The Computer System Usability Questionnaire, developed by IBM, contains 19 statements to which participants respond using a 7-point scale. (vgl. Shneiderman & Plaisant 2009)
Cultural Model	The consolidated cultural model shows the common aspects of culture that pertain across the customer population. It is an index of issues that matter to the people doing the work - what they care about, how they think about themselves and the jobs they do, and what constraints and policy they operate under. (Beyer & Holtzblatt 1997)
DECIDE Framework	A framework to guide evaluation based on six items: Determine the goals, Explore the questions, Choose the evaluation approach and methods, Identify the practical issues, Decide how to deal with the ethical issues, Evaluate, analyse, interpret, and present the data. (Preece et al. 2007)
Dialogue Management Reference Model	The model presents an abstract overview of the information flow during human-computer dialogue showing the relationship between language processing in the

	human and language processing in the computer. (Sisson 1986)
Diaries	In this technique, participants are asked to write a diary of their activities on a regular basis, e.g. what they did, when they did it, what they found hard or easy, and what their reactions were to the situation. (Preece et al. 2007)
Effektivität	Die Genauigkeit und Vollständigkeit, mit der Benutzer ein bestimmtes Ziel erreichen. (ISO 9241-11:1998)
Effizienz	Der im Verhältnis zur Genauigkeit und Vollständigkeit eingesetzte Aufwand, mit dem Benutzer ein bestimmtes Ziel erreichen. (ISO 9241-11:1998)
Ergonomie	Wissenschaftliche Disziplin, die sich mit dem Verständnis der Wechselwirkungen zwischen menschlichen und anderen Elementen eines Systems befasst, und der Berufszweig, der Theorie, Prinzipien, Daten und Methoden auf die Gestaltung von Arbeitssystemen anwendet mit dem Ziel, das Wohlbefinden des Menschen und die Leistung des Gesamtsystems zu optimieren. (ISO 9241-210:2010)
Essential Use Case	An essential use case is structured narrative, expressed in the language of the application domain and of users, comprising a simplified, generalized, abstract, technology-free and implementation-independent description of one task or interaction that is complete, meaningful, and well-defined from the point of view of users in some role or roles in relation to a system and that embodies the purpose or intentions underlying the interaction. (Constantine & Lockwood 1999)
Ethnography	Ethnography has traditionally been used in the social sciences to uncover the social organization of activities, and hence to understand work. A large part of most ethnographic studies is direct observation, but interviews, questionnaires, and studying artifacts used in the activities also feature in many ethnographic studies. (Preece et al. 2007)
Feature	A service the system provides to fulfill one or more stakeholder needs. (Leffingwell & Widrig 2003)
Field Studies	The distinguishing feature of field studies is that they are done in natural settings with the aim of understanding what people do naturally and how products mediate their activities. (Preece et al. 2007)
Fitts' Law	Fitts' Law (1954) predicts the time it takes to reach a target using a pointing device. (Preece et al. 2007)

Flow Model	The flow model shows the scope of the work domain a project intends to address and shows how the work the project is focused on fits into the customers' larger work practice. Flow model consolidation reveals the common structure that underlies all the different ways organizations define jobs. It does this by using roles as the essential element of work practice on which to base consolidation. (Beyer & Holtzblatt 1997)
Focal Use Case	Focal use cases are the focus of attention around which the user interface or some portion of the user interface will be organized. Focal use cases for the entire system embody the main, central, important, or representative uses of the system. (Constantine & Lockwood 1999)
Formal Usability Inspection	The experts hold a courtroom-style meeting, with a moderator or judge, to present the interface and to discuss its merits and weaknesses. (Shneiderman & Plaisant 2009)
Fully Dressed Use Case ...	A fully dressed use case is written with one of the full templates, identifying actors, scope, level, trigger condition, precondition, and the rest of the template header information, plus project annotation information. (Cockburn 2000)
Gebrauchstauglichkeit	Das Ausmaß, in dem ein Produkt durch bestimmte Benutzer in einem Nutzungskontext genutzt werden kann, um bestimmte Ziele effektiv, effizient und zufriedenstellend zu erreichen. (ISO 9241-11:1998)
GOMS Model	The GOMS model was developed in the early 1980s by Stu Card, Tom Moran, and Alan Newell. It was an attempt to model the knowledge and cognitive processes involved when users interact with systems. The term „GOMS“ is an acronym which stands for <i>goals</i> , <i>operators</i> , <i>methods</i> , and <i>selection rules</i> . (Preece et al. 2007)
Guidelines Review	The interface is checked for conformance with the organizational or other guidelines document. (Shneiderman & Plaisant 2009)
Heuristic Evaluation	Heuristic evaluation is a usability inspection technique first developed by Jakob Nielsen and his colleagues, in which experts, guided by a set of usability principles known as heuristics, evaluate whether user-interface elements, such as dialog boxes, menus, navigation structure, online help, etc., conform to the principles. (Preece et al. 2007)

HIP	The human information processing model from the perspective of (a) the psychological knowledge at that time and (b) a task-based approach to human-computer interaction. (vgl. Benyon et al. 2005)
Human Factors	siehe <i>Ergonomie</i>
Human-Centered Design ..	Approach to systems design and development that aims to make interactive Systems more usable by focusing on the use of the system and applying human factors/ergonomics and usability knowledge and techniques. (ISO 9241-210:2010)
IFIP	International Federation for Information Processing
IMPACT Model	An acronym to encapsulate the decisions to be made in planning: Intention, Metrics (and measures), People, Activities, Contexts and Technologies. (Benyon et al. 2005)
Information Scenario	These are elaborations of activity scenarios that provide details about the information that the system will provide to users. (Rosson & Carroll 2002)
Interaction Framework ...	The interaction framework by Abowd and Beale attempts a more realistic description of interaction by including the system explicitly, and breaks it into four main components: the System, the User, the Input and the Output. (vgl. Dix et al. 2004)
Interaction Scenario	These scenarios describe the details of user action and feedback. Each interaction scenario is a fully specified design vision: the users and task(s) being supported, the information needed to carry out the task, the actions the users take to interact with the task information, and the responses the system provides to users' actions. (Rosson & Carroll 2002)
Interaktives System	Kombination von Hardware, Software und/oder Dienstleistungen, die Eingaben von einem Benutzer empfängt und Ausgaben zu einem Benutzer übermittelt. (ISO 9241-210:2010)
Keystroke Level Model ...	The keystroke level model differs from the GOMS model in that it provides actual numerical predictions of user performance. Tasks can be compared in terms of the time it takes to perform them when using different strategies. (Preece et al. 2007)

Main Success Scenario	The one scenario written in full, from trigger to completion, and includes goal delivery and any bookkeeping that happens after. It is a typical and illustrative success scenario, even though it may not be the only success path. (Cockburn 2000)
Mental Model	A mental representation that people use to organize their experience about themselves, others, the environment and the things with which they interact. The functional role of mental models is to provide predictive and explanatory power for understanding these phenomena. (Preece et al. 1994)
Mock-up	Term used interchangeably with prototype. (Holtzblatt et al. 2005)
Mood Board	A mood board is a pinboard or poster design that gather visual stimuli of how a person feel composing photographs, images, colours, textures, shapes, headlines from newspapers or magazines, quotations from people, pieces of fabric and so on. (vgl. Benyon et al. 2005)
MOT	Metaphors of Human Thinking (MOT): The experts conduct an inspection that focuses on how users think when interacting with an interface. (Shneiderman & Plaisant 2009)
Navigation Map	The context navigation map represents the overall architecture of the user interface by modeling the relationships among interaction contexts. (Constantine & Lockwood 1999)
Nutzungserlebnis	Wahrnehmungen und Reaktionen (z.B. Gefühle, Verhalten, Fähigkeiten, Einstellungen) einer Person, die aus der tatsächlichen und/oder erwarteten Benutzung eines Produktes, eines Systems oder einer Dienstleistung resultieren. (ISO 9241-210:2010)
Nutzungskontext	Die Benutzer, Arbeitsaufgaben, Arbeitsmittel (Hardware, Software und Materialien) sowie die physische und soziale Umgebung, in der das Produkt genutzt wird. (ISO 9241-11:1998)
Nutzungsorientierte Gestaltung	siehe <i>Human-Centered Design</i>
Observation	Observation is a useful data gathering technique at any stage during product development. Observation conducted in evaluation may be used to investigate how well the developing prototype supports these tasks and goals. (vgl. Preece et al. 2007)

Paper-based Prototype ...	A paper based prototype is a paper representation of a product. Constructed out of notes, various other pieces of paper, or any other materials, it allows to test the design with the user interactively. Everything in the prototype needs to be moveable and changeable because user content will be added, interface structure changed, and otherwise modified in response to the customer trying to use it. (vgl. Holtzblatt et al. 2005)
Participatory Heuristic Evaluation	Does the same as heuristic evaluation using „real users“ instead of usability experts. (vgl. Preece et al. 2007)
Persona	Personae are rich descriptions of typical users of the product under development that the designers can focus on and design the product for. They don't describe real people, but are synthesized from a number of real users who have been involved in data gathering exercises. (Preece et al. 2007)
Persona Use Case	The persona use case consists of a description of the user group and the variations depending on the educational status and their context of work. It also includes impacts of the work environment, regional distinctions, and legal policies. Thereby, the persona use case gives a holistic image of one potential group of user of the future system, their work environment, their daily business and their daily tasks. (Nebe et al. 2006)
Phase	Eine Phase bezeichnet einen zeitlichen Abschnitt innerhalb eines Projektes. Dieser ist sachlich gegenüber den anderen Abschnitten getrennt und setzt sich aus darin stattfindenden Aktivitäten zusammen. (Winter 2005)
Physical Model	Individual models show how the place is structured, how it is organized to support work, and how people and things move through the space in the course of getting work done. The consolidated physical models show the common physical structure across the customer population and the key variants that a system will have to deal with. (Beyer & Holtzblatt 1997)
Pluralistic Walkthrough ..	Pluralistic walkthroughs are another type of walkthrough in which users, developers and usability experts work together to step through a [task] scenario, discussing usability issues associated with dialog elements involved in the scenario steps. (Preece et al. 2007)

Problem Scenario	A story of current practice developed to reveal aspects of the stakeholders and their activities that have implications for design. (Rosson & Carroll 2002)
Produkt	Derjenige Teil der Arbeitsmittel (Hardware, Software und Materialien), für den Gebrauchstauglichkeit spezifiziert oder evaluiert wird. (ISO 9241-11:1998)
Prototyp	Darstellung der Gesamtheit oder eines Teils eines interaktiven Systems, die, gegebenenfalls mit Einschränkung, zur Analyse, Gestaltung und Bewertung verwendet werden kann. (ISO 9241-210:2010)
PSSUQ	The Post-Study System Usability Questionnaire, developed by IBM, has 48 items that focus on system usefulness, information quality, and interface quality. (vgl. Shneiderman & Plaisant 2009)
QUIS	The Questionnaire for User Interaction Satisfaction (QUIS), developed by the University of Maryland Human-Computer Interaction Laboratory, is one of the most widely used questionnaires for evaluating interfaces. (Preece et al. 2007)
Requirement	A capability needed by the user to solve a problem to achieve an objective. (Leffingwell & Widrig 2003)
Screen Design Standards .	A set of product-specific standards and conventions for all aspects, based on any mandated industry and/or corporate standards, the data generated in the Requirements Analysis phase, and the product-specific Conceptual Model Design. Screen Design Standards will ensure coherence and consistency - the foundations of usability - across the user interface. (Mayhew 1999)
Semantic Differential	Semantic differential scales explore a range of bipolar attitudes about a particular item. Each pair of attitudes is represented as a pair of adjectives. The participant is asked to place a cross in one of a number of positions between the two extremes to indicate agreement with the poles. (vgl. Preece et al. 2007)
Sequence Model	Consolidated sequence models bring together many instances of many individuals accomplishing the same task, revealing what is important to doing the work: what needs to be done, the order and strategy for doing it, and all the different motivations driving specific actions. (Beyer & Holtzblatt 1997)

Seven-Stages	Don Norman's seven-stages of interaction model is the most influential in Human-Computer Interaction. The interactive cycle can be divided into two major phases: execution and evaluation. These can then be subdivided into further stages, seven in all. (vgl. Dix et al. 2004)
Sketch	Sketches are quickly hand-drawn captures of inspirations, ideas and thoughts. (vgl. Benyon et al. 2005)
Snapshot	Snapshots in a design are represented key moments in an interaction and are particularly useful for exploring the impact of a certain style or design. (vgl. Benyon et al. 2005)
Stakeholder	Stakeholders are people or organizations who will be affected by the system and who have a direct or indirect influence on the system requirements. (Kotonya & Sommerville 1998)
Storyboard	A storyboard, in the style of film making, is a simple cartoon-like structure in which key moments from the interactive experience are represented so that you get a feel for the „flow“ of the experience. (vgl. Benyon et al. 2005)
Subfunction Use Case	A subfunction use case satisfies a partial goal of a user-goal use case or of another subfunction; its steps are lower-level subfunctions. (Cockburn 2000)
SUMI	Software Usability Measurement Inventory (SUMI) is a internationally standardized questionnaire with 50 questions, each of which is answered with „agree“, „undecided“ or „disagree“. (vgl. Preece et al. 1994)
Summary-Level Use Case .	A summary-level use case is one that takes multiple user-goal sessions to complete, possibly weeks, months, or years. Its sub use cases can be any level of use case. (Cockburn 2000)
SUS	The System Usability Scale, developed by John Brooke, consists of 10 statements with which users rate their agreement. Half of the questions are positively worded and the other half are negatively worded. (vgl. Shneiderman & Plaisant 2009)
System Use Case	The phrase system use case is a shortcut indicating that the use case emphasizes the operations of the computer or mechanical rather than the operation of a business. It is possible to write a system use case at any goal level and at any scope, including enterprise. A system use case

	written at enterprise scope highlights the effect of the system under discussion on the behaviour of the enterprise. (Cockburn 2000)
Task Analysis	The process of investigating a problem by breaking down tasks that potential users of a system do or would do into sequences of actions and objects. (Preece et al. 1994)
Think Aloud	The technique requires people to say out loud everything that they are thinking and trying to do, so that their thought processes are externalized. (Preece et al. 2007)
Usability	siehe <i>Gebrauchstauglichkeit</i>
Usability Specification	Precise and testable statements of the usability characteristics required of a system. The intention is to specify and manage usability goals in parallel with the functional goals for the system. (Rosson & Carroll 2002)
Usability Testing	Usability testing involves measuring typical users' performance on typical task. This is generally done by noting the number and kinds of errors that the users make and recording the time that it takes them to complete the task. (Preece et al. 2007)
Use Case	A use case describes the interaction between people (or other „actors“) and devices. It is a case of how the system is used and hence needs to describe what people do and what the system does. Each use case covers many slight variations in circumstances - many concrete scenarios. (Benyon et al. 2005)
Use Case Brief	A use case brief is a one-paragraph synopsis of the use case. (Cockburn 2000)
Use Case Map	The use case map for a given problem partitions the total functionality of the system into a collection of interrelated essential use cases. By separating out distinct and meaningful interactions and showing how are related, we can construct a simpler overall model of the supported work and the capability the system should supply. (Constantine & Lockwood 1999)
User Experience	siehe <i>Nutzungserlebnis</i>
User Interface	siehe <i>Benutzungsschnittstelle</i>
User Interface Design	Detailed design of the complete product user interface based on the refined and validated Conceptual Model and Screen Design Standards documented in the product Style Guide. (Mayhew 1999)

User Need	A reflection of the business, personal, or operational problem (or opportunity) that must be addressed in order to justify consideration, purchase, or use of a new system. (Leffingwell & Widrig 2003)
User Profile	The collection of attributes for a „typical user“ is called a user profile. (Preece et al. 2007)
User Story	User Stories are the real-world experiences, ideas, anecdotes and knowledge of people. (Benyon et al. 2005)
User-Goal Use Case	A user-goal use case satisfies a particular and immediate goal of value to the primary actor. It is typically performed by one primary actor in one sitting of 2 to 20 minutes (less time if the primary actor is a computer), who can then proceed with other things. (Cockburn 2000)
Vision	A vision is a hand-drawn, graphical representation. It is a high-level story of the customers' new practice. The vision describes what is in the new environment and how it works as told from the user's point of view. It includes possible changes in technology, user interface function, underlying business rules, role definition, and process. It can also include help, training, and overall support and services. (Holtzblatt et al. 2005)
WAMMI	The Website Analysis and MeasureMent Inventory questionnaire does web-based evaluations and is available in more than a dozen languages. (vgl. Shneiderman & Plaisant 2009)
Wireframe	The wireframe is a bare-bones depiction (as the name suggests) of all the components of a page and how they fit together. (Garrett 2002)
Ziel	Ein angestrebtes Arbeitsergebnis. (ISO 9241-11:1998)
Zufriedenstellung	Freiheit von Beeinträchtigungen und positive Einstellung gegenüber der Nutzung des Produkts. (ISO 9241-11:1998)

Danksagung

Ein gesprochenes „Danke“ ist wesentlich mehr Wert als ein geschriebenes, weshalb ich mich auch bei allen an meiner Masterthesis beteiligten Personen persönliche bedanken werde. Dennoch möchte ich diese Personen, die mich in der vergangenen Zeit unterstützt und an mich geglaubt haben, an dieser Stelle kurz erwähnen.

Zunächst möchte ich meinem Erstprüfer Prof. Dr. Gerhard Plakmann danken, für die Unterstützung bei der Erstellung meiner Masterthesis und dem Erwecken meines Interesses den Menschen zu betrachten und bei der Entwicklung gebrauchstauglicher Produkte Einfluss zu nehmen. Seine Vorlesungen, die gemeinsamen Gespräche und konstruktiven Ratschläge konnten mein Wissen stets erweitern, so dass ich mich sehr gut auf das kommende Berufsleben vorbereitet fühle.

Danken möchte ich ebenfalls meinem Zweitprüfer Dr. Karsten Nebe, für die Betreuung und Beurteilung dieser Arbeit. Seine Dissertation und sein konstruktives Feedback konnten erheblich dazu beitragen, das umfangreiche Thema der Integration von Usability Engineering und Software Engineering zu erfassen und in angemessener Weise zu bearbeiten.

Ebenso gilt mein Dank Dirk Zimmermann, Thomas Geis, Prof. Dr. Mario Winter und Prof. Dr. Marion Halfmann für die Beantwortung meiner zahlreichen Fragen, die maßgeblich zum Gelingen dieser Arbeit beigetragen haben. Danke Dirk für deinen Vortrag zum Berufsbild des Usability Engineers, der richtungsweisend für die Findung meiner beruflichen Vorstellungen war.

Meinen Freunden, Kommilitonen und Bekannten danke ich für ihre Freundschaft und ihre allzeit ermutigenden Worte. Mein besonderer Dank gilt Christiane Grünloh für unsere fachlichen Diskussionen, ihr offenes Ohr bei Problemen und Schwierigkeiten, sowie der Korrektur dieser Arbeit.

Allen die nicht namentlich Erwähnung fanden, aber zum Gelingen dieser Arbeit beigetragen haben, sei ebenso ein Dank ausgesprochen.

Mein größter Dank gilt meiner Familie und meiner Freundin Gianna für die allzeit liebevolle Unterstützung, ihr Verständnis, ihren Halt und den Glauben an mich.

DANKE!

Eidesstattliche Erklärung

Ich versichere, die von mir vorgelegte Arbeit selbständig verfasst zu haben.

Alle Stellen, die wörtlich oder sinngemäß aus veröffentlichten oder nicht veröffentlichten Arbeiten anderer entnommen sind, habe ich als entnommen kenntlich gemacht. Sämtliche Quellen und Hilfsmittel, die ich für die Arbeit benutzt habe, sind angegeben.

Die Arbeit hat mit gleichem Inhalt beziehungsweise in wesentlichen Teilen noch keiner anderen Prüfungsbehörde vorgelegen.

Nümbrecht, 26. Juli 2010

Holger Fischer